# Measuring and Disrupting Malware Distribution Networks: An Interdisciplinary Approach.

*Colin C. Ife*

A dissertation submitted in partial fulfillment

of the requirements for the degree of

**Doctor of Philosophy**

of

**University College London**.

Department of Security and Crime Science

University College London

February 12, 2021

I, Colin C. Ife, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

# Abstract

Malware Delivery Networks (MDNs) are networks of webpages, servers, computers, and computer files that are used by cybercriminals to proliferate malicious software (or malware) onto victim machines.

The business of malware delivery is a complex and multifaceted one that has become increasingly profitable over the last few years. Due to the ongoing *arms race* between cybercriminals and the security community, cybercriminals are constantly evolving and streamlining their techniques to beat security countermeasures and avoid disruption to their operations, such as by security researchers infiltrating their botnet operations, or law enforcement taking down their infrastructures and arresting those involved. So far, the research community has conducted insightful but isolated studies into the different facets of malicious file distribution. Hence, only a limited picture of the malicious file delivery ecosystem has been provided thus far, leaving many questions unanswered.

Using a data-driven and interdisciplinary approach, the purpose of this research is twofold. One, to study and measure the malicious file delivery ecosystem, bringing prior research into context, and to understand precisely how these malware operations respond to security and law enforcement intervention. And two, taking into account the overlapping research efforts of the information security and crime science communities towards preventing cybercrime, this research aims to identify mitigation strategies and intervention points to disrupt this criminal economy more effectively.

# Impact Statement

The research conducted herein is focused on cybersecurity and cybercrime prevention, particularly in relation to malware delivery and botnet operations. Several contributions with diverse impacts are derived from this work.

First, multiple analytical methodologies are devised using big (security) data – specifically, download metadata – to build graph representations that mirror the real-world networks used to deliver suspicious and unwanted software on the Web. These methodologies enable (i) in-depth cross-sectional and longitudinal analysis of big (security) data at different granularities (e.g., infrastructure-level, operation-level, ecosystem-level); and (ii) identification of structurally critical and stable nodes within such graph networks, mirroring key components in malicious file delivery infrastructures online (files, domains, IPs). These analytical methodologies and the intelligence derived from them can be used and acted upon by various stakeholders. For example, law enforcement, security companies, and researchers around the world may use them to identify weak points in a malware delivery or botnet operation for effective takedown counter-operations. The source code for these methods have been released publicly for others to use and build upon.

Second, detailed analyses are conducted which denude the structures, workings, evolution, and distinct behaviours of the malicious file delivery ecosystem and individual malware delivery operations. Many of the findings are novel, while others confirm findings of other works and put them into a broader context. In addition, a comprehensive survey of cybercrime research from the perspectives of information security and environmental criminology is conducted. This study is one of the first of its kind, generating several new insights into cybercrime analysis and

prevention techniques, and helps to establish a new, complementary research direction between information security and crime science. This study also contributes to other academic and non-academic fields, such as the proposal of a novel concept of *cyberplace* – the digital analogue of interactional environments in the real world – being relevant to regional and geographical sciences, computer science, urban technology, and the legal sector, to name a few. The full ramifications of this concept are yet to be realised. The knowledge generated from these analyses benefit both the academic and non-academic communities, contributing to the body of knowledge for teaching and further research, and providing a synthesised knowledge base for stakeholders with an interest in cybercrime analysis and prevention, such as security specialists, sociotechnical system designers, and public policy practitioners.

Finally, novel uses of existing frameworks for crime prevention are considered to devise new cybercrime countermeasures. Some new, proof-of-concept countermeasures are proposed using said frameworks. The most obvious beneficiaries of such frameworks and proposed countermeasures are the security community and law enforcement sector, who may begin to apply, evaluate, and refine them. However, more generally, academic and non-academic stakeholders may work together to test and refine these proposed frameworks and proof-of-concept countermeasures, particularly through the use of evidence-based approaches and action research models.

# Acknowledgements

*"I returned, and saw under the sun, that the race is not to the swift, nor the battle to the strong, neither yet bread to the wise, nor yet riches to men of understanding, nor yet favour to men of skill; but time and chance happeneth to them all."*

*~ Ecclesiastes 9:11*

An apt summary of my doctoral journey indeed! When I first embarked on this journey, I envisioned the recipe for success as being the components of a simple equation: a couple of years of learning, questioning, experimentation, networking, long days, and perhaps some long nights as well. Were it so easy. Rather, for me, facing the most unique, untimely, and significant challenges of my life during this degree, I soon realised that there were far more important ingredients for success: relationships, encouragement, perseverance, inner-strength, self-confidence, patience, faith, serendipity, and, at times, simply undeserved and unmerited favour. After these last four years, I am nothing but humbled and grateful.

First and foremost, I thank God Almighty for upholding me every single day, and making me a better person ending this chapter than when I began.

I thank my family and beloved ones for their continued love and support.

I thank the love of my life, Onyinye, for her unwavering support long before us ever becoming a couple – clearly I chose well.

I thank Pastor Uzor Ndekwu and Jesus Sanctuary Ministries for their spiritual and financial support, especially when things were at their most difficult – God will surely make bountiful your investment of love.

I thank my former Director of Studies, Professor Hervé Borrion, for not allowing me to give up when I was at my lowest point – I remember the very conversation and your efforts to bring me back up, and I will never forget it.
I thank my second supervisor, Professor Steven Murdoch, for all his advice, guidance, efforts, and the opportunities he afforded me to make me a better researcher – I am nothing but grateful to have worked with you these last five years.
I thank my first supervisor and academic shepherd, Professor Gianluca Stringhini, for far too much to even begin to list. In short, you went above and beyond the duties and expectations of any doctoral supervisor and are living proof that angels walk among us. My prayer is that your labour of love, mentorship, and care towards me will be a memorial before the Almighty God forever.
Moreover, I thank my co-authors, Dr Yun Shen and Dr Toby Davies, my pre-doctoral mentor, Dr Sylvester Eyong, and the long list of colleagues, friends, brethren, reviewers, and professors too numerous to name who gave me their time, concern, advice, encouragement, and the pleasure of working with them over these years – God bless you all.

And finally, I end as I began: I thank God Almighty for each and every day of this journey: past, present, and, with great excitement, future.

To Him be the glory forever.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Malware is software that is designed to carry out malicious activities on a victim's computer system, usually without the permission or knowledge of its owner. There are several main types of malware, including computer viruses, trojans, droppers, worms, and rootkits. Malware is constantly evolving in its capabilities, characteristics, and modus operandi as cybercriminals are continually seeking new ways to carry out their criminal activities while avoiding detection or disruption. Given that malware activity is involved in most technical crimes, it is recognised as one of the most severe security threats of our time.

## 1.1 Evolution of Malware Delivery

Malware delivery has undergone an impressive evolution since its inception in the 1980s, moving from being an amateur endeavour to a perfectly oiled criminal business. In pursuing larger and larger populations of victims, malware authors moved from using floppy disks as their infection vector [106] to delivering malware as attachments in spam emails [188], enticing users into opening them with social engineering [155]. Eventually, malware authors started compromising user machines without the need of explicit user interaction, by exploiting vulnerabilities in the victim browser once it visited a malicious web page (a so-called *drive-by download attack* [165]). This increase of sophistication in the malware delivery process evolved side by side with miscreants developing increasingly profitable ways of monetising their operations [136, 186, 125].

An issue with drive-by downloads is that vulnerabilities typically affect single versions of web browsers or plugins, and vendors are constantly patching them. This hardly reconciles with the need of cybercriminals to infect as many victims as possible, across a variety of software configurations, and for a long period of time. To ease the life of malware operators seeking to infect victims through drive-by downloads, the cybercrime ecosystem came up with *exploit kits* (EKs) [98] – software packages that contain exploits for multiple vulnerabilities. Exploit kits are able to fingerprint the victim system and deliver an appropriate exploit that is able to compromise the system [75]. Malware operators can therefore purchase an exploit kit (or rent one as-a-service [98]) and efficiently infect victims.

In a further attempt to streamline malware delivery and lower the entry bar for criminals wanting to undertake a career in malware, the cybercrime ecosystem introduced *pay-per-install* (PPI) schemes [47]. In these operations, a specialised actor sets up a network of infected computers (commonly known as a botnet [21]); the malware on these victim computers do not perform any activity other than downloading additional components. Customers of PPI services can then pay their operator to install malware of their choice on a certain number of victim computers. The widespread adoption of exploit kits and pay-per-install services has created a complex underground ecosystem, in which different cybercriminal actors trade services with each other, and each specialise in a particular step in the criminal operation.

More recently, researchers uncovered a parallel economy that shares many traits with malware, while being largely controlled by different actors: the one of *potentially unwanted programs (PUPs)* [129, 127, 200]. This category of programs include software that is not willingly installed by users, and that typically is an annoyance more than a direct threat to the safety of victims — examples include adware and browser toolbars. Research showed that while malware delivery mostly happens through drive-by downloads, PUP victims are usually tricked into installing a downloader, or *dropper*, through social engineering [127]. After such a dropper is installed, additional components are dropped through a PPI service [200].

To complete this already variegated picture of malicious software distribution, the cybercrime ecosystem has developed multiple techniques to make takedowns by law enforcement and detection by security companies more difficult. Miscreants use *Fast Flux* [109] techniques, in which the Internet Protocol (IP) address associated with a certain domain is changed very quickly. Similarly, to make it difficult to identify DNS domains involved in an illicit operation, cybercriminals use *Domain Generation Algorithms (DGAs)* [27], which algorithmically changes Domain Name System (DNS) domains constantly, allowing malicious hosts to know which domain to contact at any time. Finally, malicious files are constantly changed to avoid easy detection, by using techniques known as *polymorphism* [31], while also employing various other anti-research techniques to fool security researchers and their analysis environments.

## 1.2 Mitigating the Threat

To defend against the continuous threat presented by malware, the security community is constantly working to improve systems security: identifying and fixing system vulnerabilities, developing more secure operating systems, discovering new intrusion strategies used by malicious actors, and developing better detection systems to block cyber threats such as malware. However, once those systems are breached and malware is installed onto them, the strategic focus of security must turn to more reactive strategies. This is because these devices can be assimilated into *botnets* – networks of infected computers – by having the malware establish a communication channel with the botnet operator's command-and-control (C&C) servers [70]. Once assimilated, this army of bots may be weaponised to commit further cybercrimes, such as distributed denial-of-service attacks against a target server, or mass-encryption of the victim devices, denying access to them (especially if they are critical infrastructure). As such, the priority for the security community becomes effecting *botnet takedown* counter-operations (which are taxonomised in Section 2.6), disinfecting the devices that were assimilated, and, if possible, arresting and prosecuting the perpetrators involved. Clearly, malware delivery is the

necessary precursor to building a botnet. However, the challenge of identifying effective intervention points in these malicious operations remains [150].

At the same time, the security community has raised several questions over the efficacy of botnet takedown operations [66, 79, 181, 83]. Given the complexities within the malware delivery process (and cybercriminal operations more generally), it is unsurprising that the security community has leveraged concepts and techniques from other fields in the hope of analysing and disrupting these operations more effectively [191]. For instance, the attack tree [176] and the cyber kill chain [114] are just two, commonly-used models to understand cyber attack sequences. However, these models are actually underpinned by extradisciplinary techniques and concepts, such as fault tree analysis from electronics engineering, or the original kill chain from the military context. Likewise, several studies into cybercriminal operations have uncovered the undeniable role of profit, business partnerships, and outsourcing in such malicious activities [201, 192, 160, 200, 127]. These studies highlight the need for the economic and business perspectives to understand cybercriminal ecosystems more profoundly and identify pressure points in such operations. More recently, the security community has begun to consider models and frameworks from fields such as environmental criminology, which are used to analyse and mitigate crime in the real world [191, 139, 135, 56]. Such fields are already interdisciplinary in nature, combining contributions from criminology, psychology, economics, geography, mathematics, and computer science to study and control crime. These are just a few examples of extradisciplinary contributions to cybersecurity, demonstrating the continued need for interdisciplinary research and collaboration to mitigate malware delivery operations, and cybercrime more generally.

## 1.3 Research Scope and Methodology

The research community has so far studied the different facets of malicious file distribution in isolation: malware prevalence, PUP prevalence, the use of pay-per-install schemes, etc. While these studies are very insightful in understanding

specific phenomena, they do not provide a full view of the malicious file delivery ecosystem, leaving many questions unanswered. E.g., what does the malicious file delivery ecosystem look like? Are there differences between the network infrastructures used to download PUP and malware? And, how do these infrastructures evolve over time? On the effects of takedown operations, how do malicious operations respond to botnet takedowns? Do they subside? Or, do they move their infrastructure elsewhere, or change their modus operandi? Furthermore, existing studies mapping the actors in cybercriminal ecosystems are few, with none looking at the file delivery ecosystem specifically, but focusing on other elements of the cybercrime pipeline, such as spam delivery and its monetary conversion [192, 136].

Therefore, utilising a data-driven approach, the primary objective of my research is to measure malicious file delivery networks comprehensively, understanding their structures and how they respond to takedown initiatives. To this end, I first conduct a measurement study of the malware and PUP delivery ecosystem on the Web. Second, I conduct a measurement study of the evolution of specific malicious file delivery operations that face takedown counter-operations. Both of these studies involve processing and analysing download telemetry collected over a year.

The secondary objective of my research is to identify better approaches to disrupting malware delivery networks. This is accomplished in two stages. First, through measurement studies, I seek to devise methodologies to identify important nodes in malware delivery networks, which may serve as effective intervention points for disrupting this criminal economy. More generally, these analytical methodologies should be applicable using data that is collected at any time. Second, I investigate cybersecurity interventions and the processes used to derive them from an interdisciplinary approach, i.e., from the information security perspective and the environmental criminology one. This is to identify opportunities to synthesise knowledge and frameworks from both fields so as to mitigate cybercriminal operations more effectively. More broadly, I not only consider the problem of malware delivery, but other malicious activities as well (Dark market solicitation, cryptocurrency crime, cyber fraud, etc). In the interest of identifying new and innovative

solutions to the malware delivery problem, this makes sense: malware delivery operations regularly rely on or lead to other forms of criminal activity, such as engaging malware- and crimeware-as-a-service providers on Dark markets to setup botnet operations, or leveraging botnets to mine cryptocurrencies, operate clickjacking operations, or implement spamming operations for further nefarious activities. As such, considering mitigations for other forms of cybercrime could lend itself useful to disrupting the complex and composite malware value chain.

## 1.4 Contribution

The contribution of this thesis is encapsulated within three studies, each of which is assigned its own chapter. In the first contribution of this thesis, I conduct a measurement study of the entire malicious file delivery ecosystem on the Web. This is to put other research on isolated aspects of malware delivery into context and answer key questions, such as what the malicious file delivery ecosystem looks like, whether there are differences in infrastructures that deliver different types of unwanted software, and how these infrastructures evolve over time? Using download metadata provided by Symantec, a novel methodology is devised to analyse malware delivery networks cross-sectionally (a snapshot of activity) and longitudinally, and identify various weak points in these criminal operations. Furthermore, this work provides the security community answers to key questions regarding the structure and workings of various aspects of this malicious ecosystem.

In the second contribution of this thesis, and as a natural extension to the first, I conduct a measurement study on the evolution of three malware delivery operations that are targeted for takedown by law enforcement and security companies. This is to establish precisely how different malware delivery operations respond to takedown counter-operations, what we can learn from such behaviours, and how such knowledge can be incorporated into future takedown strategies. Through this work, a novel methodology is devised to analyse file delivery operations longitudinally and in great depth. This methodology is not limited to any specific family or type of software, neither is it limited to small-scale studies. Furthermore, this work

gives the security community deep insight into the different structures, dynamics, business relationships, and behaviours of the studied malware operations – some of which have never before been documented in security literature or industry reports. The analysis code used for both measurement studies is publicly released for other researchers, analysts, and practitioners to use.

In the third and final contribution of this thesis, I conduct an extensive survey of the cybercrime literature from the perspectives of information security and environmental criminology. In this survey, I draw parallels and explicit links between cybercrime research from information security and the theories and practicies of environmental criminology. Next, I demonstrate how security researchers and practitioners could apply frameworks from environmental criminology to generate cybercrime countermeasures. Using such frameworks, I propose some new cybercrime countermeasures as proofs-of-concept. Finally, I propose a novel concept of *cyberplace* – the digital analogy to environments wherein crimes and malicious behaviours are committed in the real world. Devising such a concept is recognised in the literature as necessary to facilitate the transfer of some important environmental criminology theories and practices [139].

## 1.5 Thesis Structure

The rest of this thesis is structured as follows: in Chapter 2, I discuss the fundamental concepts, technologies, and techniques used in malware delivery and in takedown operations. In Chapter 3, I describe the data sources used in my studies. In Chapter 4, I present a longitudinal measurement study of the malicious file distribution ecosystem. In Chapter 5, I present an evolutionary study of malware delivery operations that suffered takedown attempts. In Chapter 6, I present a survey of cybercrime literature from the information security and environmental criminology perspectives, identifying how environmental criminology could be applied to cybercrime prevention, and what further work is required. In Chapter 7, I summarise and discuss the contributions presented in this thesis, while in Chapter 8, I give recommendations for future work.

# Chapter 2

# Fundamentals of Malware Delivery and Related Work

The delivery of malicious files on the Internet involves two main aspects: the *malicious payloads* themselves and the *network infrastructures* used by cybercriminals to install them onto computers. This section aims to provide an overview of the fundamentals of malware delivery. Namely, I will discuss the key concepts and representative research in relation to malicious payloads, the types of payload delivery techniques used by cybercriminals, and the techniques and technologies they use to enhance network resiliency. Complex relationships and interactions arise out of the many variables in malware delivery. As such, I will also discuss research on the resulting dropper networks that form, and studies on measuring the actors involved in cybercrime. I will then discuss work relating to botnet takedowns – the main strategy for disrupting malware delivery networks. Finally, I will also discuss the most recent interdisciplinary research in cybersecurity, particularly in relation to environmental criminology.

## 2.1 Malicious Payloads

Previous research has identified two main types of malicious files being delivered on the Internet: *malware* and *potentially unwanted programs (PUPs)*. Recent research has shown that malware and PUPs are different problems with separate characteristics [200, 127].

### 2.1.1 Malware

Malware has been a rising problem for over three decades. Previous research has focused on studying the ways in which malware obfuscates itself to avoid easy detection [58, 31], such as through the use of inexpensive packer software [218]. This technique of binary obfuscation is called *polymorphism*. Over the years, malware has been used for a number of reasons: sending spam emails [188], stealing banking credentials from infected computers [186, 33], and encrypting victim data and asking for a ransom [125], just to name a few.

Researchers have also identified a plethora of means in which malware is delivered: transmission through physical media [106], malicious attachments in spam emails [188], social engineering [155] (e.g., tricking a victim into downloading the malware from a malicious link), drive-by downloads [165] – the process of victim browsers being exploited after visiting a malicious web page, or viewing a malicious advertisement – or *exploit kits* [98] – software packages that contain exploits for diverse software configurations – that are hosted on compromised web content. In recent years, however, the research community has shown that prominent malware families are often downloaded by *droppers* that belong to PPI services [186, 188]. This is one of the latest distribution techniques to be developed by the cybercriminal economy.

### 2.1.2 Potentially Unwanted Programs (PUPs)

Potentially unwanted programs (PUPs) are software that contain adware, spyware and toolbars with annoying, undesirable, or undisclosed behaviours. PUPs are usually bundled with free software, or custom installers of a *wanted* program that the user gives consent to download, and are installed onto a user's machine without giving explicit opt-out choices. In most cases, these PUP track the Internet usage of users and display pop-up ads and advertisements on web pages that the users visit, promoting the installation of additional questionable content, including web browser toolbars, optimisation utilities, and other products. One worthy example is `sourceforge.net`. It terminated its "DevShare" program that delivered in-

staller bundles as part of the download that include unwanted software (e.g., Ask Toolbar, OpenCandy adware, etc.) [7].

Recent research shows that PUP is rapidly becoming an important problem. For example, two recent papers show that rogue browser extensions that contain hidden functionalities are on the rise [119, 124]. One study reported that 5% of Google users have installed browser extensions that substitute the advertisements that they see [199]. This can be particularly dangerous as rogue ad networks can be used to infect users with malware through drive-by download attacks [222].

PUP has risen to a new frontier of threats to users with its increasing prevalence in recent years. For example, researchers [199] have observed 192 deceptive Chrome extensions impacting 14 million users and more than 5% of unique daily Internet Protocol (IP) addresses accessing Google. Others [119] have found that malicious browser extensions are capable of infecting over 50 million Chrome users, highlighting that the extension abuse ecosystem, leveraging web traffic and user tracking, is considerably different from the malware ecosystem. The authors then summarised lessons from three years fighting malicious extensions and proposed *WebEval*: a system that identifies them. WebEval used a blend of automated systems and human rules leveraging features extracted from an extension's behaviours, code base, and developer reputation to achieve a measurable detection rate of 96.5%. *Hulk* [124] is another dynamic analysis system that has been introduced to detect malicious behaviour in browser extensions by monitoring their execution and corresponding network activity.

Another study [137] filtered over 26.8 million network traces observed from dynamic malware execution, measuring and comparing the use of domains between malware and PUPs. It confirmed that PUPs were on the rise, and that they relied on stable Domain Name System (DNS) and IP infrastructure, with several hundred thousand PUP samples using the same network infrastructure over a year.

Measuring PUP prevalence more generally, one study [200] provided a systematic study of PUP prevalence and its distribution through *commercial* pay-per-install (PPI) services, mainly focusing on four major downloaders from Amonetize,

InstallMonetizer, OpenCandy and Outbrowse. It was reported that commercial PPIs drive over 60 million download attempts per week and knowingly attempt to evade user protections (e.g., antivirus software). Another study [127] also measured PUP prevalence and its distribution through PPI services. However, in this work, the authors identified dominant PUP publisher names from code signing certificates. The authors claim that the fundamental difference between malware and PUP is the distribution mechanism. They argue that malware distribution is dominated by *silent* installation through vulnerability exploitation, while PUP is installed with the consent of the users (either consciously or unconsciously).

Understanding the relationships and relative scales between malware and PUP plays an important role in this thesis. In particular, in Chapter 4, I devise a methodology to measure and compare the structures, sizes, proliferation, and lifespans of malware and PUP delivery infrastructures on the Web that target desktop devices. Likewise, I investigate shared distribution infrastructures between the two types of unwanted software to uncover how commonly such arrangements exist. This is to give the security community a deeper understanding on the workings and relationships within such infrastructures, and better perspective on the relative scales of the two unwanted software problems.

## 2.2 Payload Delivery Techniques

The research community has identified two main infrastructures that are used by cybercriminals to deliver malware: *exploit kits* and *pay-per-install services*.

### 2.2.1 Exploit Kits

Exploit kits have been used for many years to spread malware. In a nutshell, exploit kits collect a large number of exploits targeting many versions of operating systems, browsers, and browser plugins to make sure that criminals can infect as many victim computers as possible [98].

One of the earliest exploit kits is MPack, a PHP-based kit released in late 2006 [98]. The main functionality of these kits is to gather information on the victim machine (otherwise known as "fingerprinting"), find vulnerabilities within it

and determine the appropriate exploit, and finally deliver the exploit (e.g., drive-by downloads) and execute the malicious payload. The process of becoming exploited by one of these kits, in general, follows these steps: a victim visits a compromised website, then is redirected to several intermediate servers, and finally lands on a host with an exploit kit. The exploit kit finds a vulnerability using the information collected from the victim (i.e., fingerprinting) and consequently delivers the malicious payload.

Nowadays, exploit kits represent the state-of-the-art in automated remote-infection technology, which have evolved with the for-profit malware ecosystem. As such, several studies have been directed towards detecting exploit kits on the Web. One work [197] leveraged the inherent structural patterns in Hypertext Transfer Protocol (HTTP) traffic to classify exploit kit instances. The proposed system captured these interactions in a "tree-like" form, and models the detection process as a subtree similarity search problem. Another study [88] surveyed a wide range of 30 real-world exploit kits and introduced the *EKHUNTER* system. This system automatically detects the presence of exploit kit vulnerabilities and compromises both the integrity of a fielded exploit kit, and even the identity of the kit operator. A third work [98] centered around the malware installed upon a successful browser exploit, and investigated the emergence of the exploit-as-a-service model for drive-by browser compromise. This is achieved by analysing over 10,000 distinct binaries extracted from 77,000 malicious uniform resource locators (URLs). This study showed that 9 exploit kits, though a small number, account for 92% of the malicious URLs in their dataset, 29% of which belong to the Blackhole exploit kit. A static analysis system, *PExy*, was designed in another work [75], which extracts the set of URL parameters and user agents from the server-side source code of an exploit kit, and recreates all the necessary conditions to trigger all exploits from an exploit kit. Note that PExy is limited by the availability of exploit-kit server-side source code.

Exploit kit activity is almost certainly captured in the dataset that I study as part of this work. However, attempting to identify such activities is beyond the scope of this thesis. This is because, as one will later find as I describe the data sources and

analysis methodology used in Chapters 3 and 4, it is infeasible to attempt to differentiate downloads from exploit kits versus downloads from other delivery vectors using network graph and metadata analysis alone. Therefore, to incorporate exploit kit detection into the methodologies devised in this thesis, one would likely require the use of a parallel analysis framework (e.g., crawling and analysing sites hosting exploit kits) or an additional source of ground truth to enrich the dataset.

### 2.2.2 Pay-Per-Install (PPI) Services

PPI services have existed for years. They originated as services to facilitate the distribution of advertisements, but have seen significant (malicious) changes over the years by centering on pushing malware and spyware to unsuspecting users [188]. A typical PPI ecosystem has three main actors: a client, a service provider, and an affiliate. A typical PPI transaction works as follows: clients (e.g., malware authors) pay PPI service providers to have their malware installed on a number of victim computers. These service providers either install the malware onto victim machines directly (i.e., using their own downloaders), or employ affiliates to distribute malware to target users (i.e., buying installs from third-parties). Once malware is successfully installed and verified by PPI clients, affiliates receive payments from the service providers.

Given the rise in this malicious use of PPIs – both commercial PPIs used to deliver malware among other types of software, and malicious PPIs that are specifically designed for malicious activity – research has been conducted in recent years to measure these services. One study [200] argues that PPIs can be divided into commercial PPIs and blackmarket PPIs. Commercial PPIs need user consent to operate while blackmarket PPIs perform silent installs on the target hosts, i.e., installations that lack the informed consent of the owner of the system. Another study [47] provided the first large-scale measurement of blackmarket PPI services in the wild. This is achieved by harvesting over a million client executables using vantage points spread across 15 countries. This work found that 12 out of 20 of the most prevalent malware families at the time employed PPI services to buy infections, confirming the previous observations that cybercriminals are commonly

using other botnets to deliver their malicious payloads. A third study [127] leveraged dropper graphs to build a *publisher graph* and identify specific publisher roles in the ecosystem. The authors tag roles (e.g., client, service provider, and affiliate) to each publisher by measuring the in-degree and out-degree of each cluster in the publisher graph. That is, publishers with both high in-degree and out-degree behave like PPI service providers; publishers with high in-degree but low out-degree are more likely advertisers; and publishers with low in-degree and high out-degree are likely affiliates.

PPI infrastructures are identified regularly throughout the work conducted as part of this thesis: first, in the measurement study of the malicious file delivery ecosystem, where I devise a technique to estimate the number of active PPIs on a single day by clustering connected effective second-level domains (e2LDs) and dropper networks (Chapter 4). And, second, in the evolutionary study of malicious delivery operations targeted for takedown, where we see the differing use of dropper networks (a core aspect of PPIs) between three different malware operations (Chapter 5). Finally, the taxonomy of countermeasures proposed for disrupting botnet and malware operations intersect with the PPI phenomenon, particularly against botnets that are monetised using this business model (Chapter 6).

## 2.3 Technologies to Enhance Network Resiliency

Cybercriminals need to make their operations resilient to takedowns. Over the years, two main technologies were developed for this purpose: *Fast Flux* and *domain generation algorithms (DGAs)*.

### 2.3.1 Fast Flux

The basic idea behind Fast Flux is to rotate between numerous IP addresses (usually from compromised machines) associated with a single fully qualified domain name. By constructing such a distributed proxy network on top of compromised machines, this technique makes malware networks more resistant to discovery and disruptive countermeasures.

The first empirical study of Fast Flux service networks (FFSNs) [109] showed that almost 30% of all domains advertised in spam were hosted via FFSNs. It also introduced several parameters (e.g., the number of unique DNS address (A) records returned in all DNS lookups, nameserver (NS) records in one single lookup, and unique ASNs for all A records[1]) to distinguish FFSNs from content delivery networks (CDNs), and several strategies to mitigate the threats. A separate work [111] involved the deployment of 240 sensors to understand global IP-usage patterns exhibited by different types of malicious and benign domains, revealing potential trends for botnet-based services. Based on these insights, the authors proposed a multi-level support-vector machine (SVM) classifier to provide fine-grained classification of fast flux domains.

The task of disentangling domains using Fast Flux from those as part of CDNs proves a difficult one, particularly without additional metadata such as DNS records. For the purposes of the studies conducted as part of this thesis, detecting the use of Fast Flux is deemed out of scope. However, observations where Fast Flux is likely used by malicious delivery infrastructures are still highlighted, particularly in the case studies analysed in Chapter 5. In any case, extending the methodologies proposed in this thesis to detect Fast Flux definitively remains a worthwhile prospect.

### 2.3.2 Domain Generation Algorithm (DGA)

Instead of using hardcoded DNS domains, malware authors employ Domain Generation Algorithm (DGA) to generate a large number of domain names as potential rendezvous points to command and control (C&C) servers, but only a portion of them are contacted to receive updates and/or commands. It makes security researchers and law enforcement unlikely to predict the next time a malware would receive an update and possibly sinkhole the C&C server address.

Significant research efforts have been directed towards detecting DGA domains and reverse-engineering the algorithms hard-coded into malware that enable

[1] A DNS address (A) record indicates the IP address for a given domain, while a DNS nameserver (NS) record indicates which DNS server is authoritative for the given domain. An Autonomous System (AS) is a collection of connected IP routing prefixes belonging to a network or collection of networks, and that are all managed by a single entity or organisation and share a common routing policy. Each system is designated a unique Autonomous System Number (ASN).

them to rendezvous with these domains. In one study [217], three metrics were proposed to differentiate a set of legitimate domain names from malicious ones – information entropy (KL-divergence), Jaccard similarity, and Levenshtein edit distance. The study showed the relative performance of each metric in different scenarios and concluded that the Jaccard measure performs the best in identifying algorithmically generated domain names. In another work [27] the *Pleiades* system was presented: a system to detect algorithmically generated domain names leveraging insight that most of the DGA-generated domain queries would result in Non-Existent Domain (NXDomain) responses, and machines from the same botnet, if employing the same DGA algorithm, would generate similar NXDomain traffic. Employing a multi-class version of the Alternating Decision Trees (ADT) learning algorithm, Pleiades successfully identified twelve DGAs (6 were previously unknown) from a large Internet service provider (ISP) network in 15 months. A third work [175] proposed *Phoenix*: a system that differentiates DGA and non-DGA domains, and attributes DGA domains to their respective botnets. Phoenix was evaluated on over 1.1 million domains, correctly distinguishing 94.8% of domains and identifying the actors behind them.

Turning to reverse-engineering research, a study [186] discussed Torpig's DGA algorithms in detail. It was shown that the Torpig DGA first generates a weekly domain name (depending on the current week and year) with a list of top-level domains (TLDs) to form potential rendezvous points. If connections to C&C servers using these weekly domain names failed, Torpig would generate another batch of potential rendezvous points using a daily domain name appended with several predefined TLDs. If all of these connections failed, Torpig would fall back to contact the domains hard-coded into its configuration file. More recently, a comprehensive measurement study [163] of 43 DGA-based malware families and their variants was carried out. By reimplementing their DGA algorithms, the authors were able to study the registration status of over 18 million DGA domains and characterised the registration behaviour of botmasters and sinkholers. The authors also

examined the effectiveness of domain mitigations and shared the full domain dataset which resulted from their work.

Again, the detection of DGA domains is out of the scope of this thesis. This is primarily due to the need for additional data sources (which I painfully learned through a number of preliminary clustering experiments). However, the use of DGA is still identified with relative certainty when observed in the case studies in Chapter 5. Just like Fast Flux, I do believe that DGA detection is a viable extension of the analysis methodologies presented in this thesis. This could be achieved either by using additional domain metadata (DNS records, WHOIS records[2]), or by implementing an unsupervised classifier and using DGA domains from online blacklists as validation data, for example.

## 2.4 Measuring Dropper Networks

Having established the core components of malware delivery, an important task for the research community has been measuring the diversity of delivery network structures and complex ecosystems that arise. To this end, several big-data studies have been carried out to understand and detect malware delivery networks at scale.

One foundational study [172] involves the large-scale analysis of 23 Windows-based malware downloaders over several years, identifying the characteristics of their binaries and the network infrastructures that they use (including PPI services). This study reports that 11 of these downloaders are active for over a year, and that 20% of malware C&C servers remain operable in the long term. However, this study stops short of measuring the interactions between different malware families and shared distribution infrastructures.

A more recent work [130] introduced a downloader-graph abstraction, which captures download activities on end hosts. The authors use this abstraction to explore the growth patterns of benign and malicious graphs. Several strong indicators of malicious activity are identified, and, subsequently, a machine learning malware detection system is built based on these insights. Building on this work, a follow-

[2]DNS records contain IP address and routing information for domains. WHOIS records contain domain ownership and contact information.

up [131] proposes *Beewolf*, a system which detects lockstep behaviours – a synchronised shift of communications from one domain to another by multiple downloader binaries – based on a file and source domain graph. The Beewolf system is used to study silent delivery campaigns involving benign software, malware, and PUP, and to assess how well it can detect suspicious activity.

Other researchers have used downloader graphs as a means for detecting malicious files. For instance, one work [24] studies a global heterogeneous malware delivery graph using both file-dropping relationships and the topology of the file distribution networks (host names, IPs). Using this topological information and content-agnostic features of different node types, a Bayesian label propagation approach is devised to identify malicious files. Around the same time, a separate work [194] proposed another malicious label propagation system for heterogeneous downloader graphs – *Marmite*. Using this system, the authors provide some insights into dropping relationships between benign software, malware, and PUP.

My work in measuring the malicious file delivery ecosystem of the Web builds on other works in this area, most of which occurred during the same time period as my own studies. Consequently, I used similar download graph techniques to investigate static malware delivery infrastructures more deeply, while, at the same time, devising new techniques for analysing different delivery infrastructures and entire operations longitudinally. As such, much of the contributions of my work is in line with giving the security community a greater understanding of this complex ecosystem and how various parts of it evolve over time.

## 2.5 Studying Malicious Actors

Another important and challenging task when studying criminal ecosystems is identifying the different actors involved in them, and mapping their relations. The studies in this area are limited, and they focus on single ecosystems instead of providing a comprehensive view of the malicious file delivery landscape.

For example, one study [47] provided an overview of the three main actors in the Pay-Per-Install ecosystem, which are PPI providers (or services), clients, and

affiliates. A later study [200] identified fifteen distinct commercial PPI networks. The authors showed that six of the fifteen PPI downloaders were merely resellers for other PPI networks, while the rest were distributors. In another study [127], a publisher relationship graph was built by leveraging file-dropping relationships and file signer information. Based on in-degree/out-degree, the authors classified publishers into PPI services, affiliates, and advertisers. Focusing on the economics of PPI services, an even more recent study [128] uses *entity graphs* to capture the network of companies and persons behind a PUP operation. This work focuses on the structures of three Spain-based PUP PPI services, identifying the actors involved in each operation, and estimating the (minimal) profit margins they achieve.

Other studies have attempted to map the relations between different cybercriminals in the spam value chain. In one work [192], a technique to fingerprint different actors involved in the spam delivery ecosystem (email harvesters, botmasters, and spammers) was developed, while another [136] looked at the spam conversion landscape, uncovering the relations between affiliate programs, Internet service providers, and payment processors.

So far, these research efforts have not systematically studied the actors in a complex ecosystem involving different types of malicious activity. It remains intriguing to answer some questions, like if different actors use different methods to distribute malicious files? What technologies an actor may adopt to operate delivery networks? etc. I address some of these questions in my own work, such as by estimating the number of active PPI services in the malicious file delivery ecosystem in Chapter 4, or by triaging specific malware delivery operations and highlighting differences in their modus operandi and business relationships in Chapter 5. Nonetheless, the task of establishing all the different actors in the malicious file delivery ecosystem (e.g., vertically integrated operation actors vs. PPI actors, operators of entire malware operations vs operators of separate crimeware-as-a-service campaigns) remains an elusive challenge that is not fully addressed in this thesis.

## 2.6 Botnet Takedowns

One final and important aspect of understanding malware delivery is the approaches the security community take to mitigate this problem. Primarily, continuous innovation in systems security is the first, proactive line of defence in protecting computers and networks from malware infections. However, once that intrusion has occurred and the malicious payloads have been delivered, the need arises to turn to more reactive intervention strategies, of which the main type is the *botnet takedown*.

Botnet takedowns are counter-operations to disrupt botnet operations and the malware delivery networks that enable their growth. Over the years, a number of different takedown strategies have been devised and implemented by law enforcement agencies (LEAs), security companies, and researchers. I summarise these as follows:

**Botnet Infiltration and Takeover.** Infiltrating a botnet is no small endeavour: it requires high technical capabilities, intelligence-gathering, and coordination [79], particularly when dealing with botnet infrastructure controlled by equally skilled, intelligent, and coordinated, malicious actors. Typically, such an operation firstly involves reconnaissance or *passive observation* – gathering intelligence on the botnet by monitoring and decoding network traffic from the infected hosts. The next stage is *infiltration*: running the botnet malware within a controlled environment (i.e., a honeypot) and analysing its internal and external workings in depth as it communicates with the rest of the malicious network. This is to acquire strategic intelligence, such as the addresses of the C&C servers and the credentials required to access them. Finally, security operatives may *takeover* these malicious networks, particularly by gaining access to the C&C servers and taking them down from within. One example of such an elaborate operation relates to the Torpig botnet, which was infiltrated by security researchers [186].

**ISP Takedown.** Another takedown approach that both public (LEAs) and private organisations (commercial companies) utilise is the *ISP takedown*. Such an approach entails a party approaching the ISP that hosts the malicious domain and requesting that they take it down for legislative reasons [50] (e.g., a court order)

or for economic reasons (e.g., otherwise other ISPs would disconnect from them). Typically, this would lead to the malicious domains and the hosted websites being deactivated, or, in some cases, the initiating party gaining control of the malicious domains from the domain registrar [133]. Unfortunately, some cybercriminals pre-empt such strategies by specifically choosing ISPs that are known to resist law enforcement pressure (so-called *bulletproof hosting services*) [25].

**DNS Sinkhole.** Particularly in the case where the initiating party can attain intermediate control of the malicious domains (or as in the case of Torpig, register DGA domains that are next to be contacted ahead of the criminal operators [186]), a common follow-up strategy is to point those C&C domains to honeypot servers and sinkhole all communications intended for them from botnet hosts. This is otherwise known as a *DNS sinkhole*, which simultaneously freezes such malicious operations while exposing victim computers within the network. This is a commonly used technique by LEAs and security companies [79].

**Seizure and Arrest.** Another takedown approach involves *physically seizing* the malicious servers, and, if possible, *arresting* and prosecuting the perpetrators. Some research has identified this to be the most effective (albeit difficult) strategy to disrupt botnet operations.

**Disinfection.** Finally, once infected machines have been identified from one of the above techniques, authorities may contact the victims and advise them on how to remove the malware from their devices. Alternatively, security companies could implement such *disinfection* campaigns remotely by pushing the removal code to the devices of their clients.

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 problem, various studies have been conducted to quantify the effects of takedowns. One study [66] examines email statistics from a medium-sized UK ISP to assess the effects of the 2008 McColo takedown on global spam volume. It found 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. A broader study [79] qualitatively analyses a set of highly publicised botnet takedown efforts between 2009-2011. It is 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 continue to make their infrastructures more resilient. The author calls for more coordination and shared knowledge within the security community to make 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 in another work [150]. 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 [151], improvements to the *rza* system are proposed 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. Another work [133] also discusses regulatory and policy solutions to botnet takedowns, particularly arguing the need for more public-private partnerships to achieve this endeavour. One study [181] 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 my 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 [83] 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 geographic region causing a diffusion of benefits and criminal activity into others.

A major focus of this thesis is in devising methods to disrupt botnet and malware delivery operations. As a result, this thesis makes significant contributions to this effect: from identifying intervention points in the malicious file delivery ecosystem (Chapter 4); to assessing the results of prior takedowns on particular malware delivery operations, seeing how they respond and evolve (Chapter 5); to considering new ways to look at cybercriminal operations in general, synthesising frameworks from other crime prevention fields, and proposing a matrix of countermeasures against botnet and malware delivery operations (Chapter 6).

## 2.7 Interdisciplinary Cybercrime Research

In this section, I review the recent shift in environmental criminology research into the digital space and cybercrime. I also briefly overview other fields of research that share an interest in defining situational or 'place' contexts in cyberspace, which is a core concept of environmental criminology theory.

### 2.7.1 Environmental Criminology and Cybercrime

In the last few decades, digital technology has undertaken an unprecedented rate of growth, culminating in it becoming a rudiment of modern society. In recent years, environmental criminologists have begun to recognise the co-dependent shift and proliferation of criminal activities in cyberspace (i.e., cybercrime), following the diffusion of criminal opportunity into the digital world. For almost two decades, discussions have been ongoing on the potential (multiplying) effect that digitisation has had on crime [97, 207]. Grabosky [97] reflects on these discussions, concluding that, though the motivations behind crime and human nature are still the same, technology has enabled an increase and diversification in criminal opportunities through anonymising technologies, transatlantic connectivity, and an absence of clear-cut boundaries for potential guardianship.

Though there has been a steady increase in studies seeking to use and evaluate the use of environmental criminology theories to model cybercrime [135, 108, 219], there is still a significant need for a proper synthesis of environmental criminology knowledge and methods with cybersecurity and cybercrime prevention paradigms. But, preceding such a synthesis, there is a more fundamental need to define situational contexts or 'cyberplaces' wherein crimes and other online activities are commissioned in cyberspace, just as (according to environmental criminology theory) situational contexts or 'places' underpin crime and other physical activities in the real world. Prior to my work in Chapter 6, formal approaches to conceptualising such 'cyberplaces' (particularly for cybercrime analysis) were almost non-existent. With that being said, one study [139] reviewed the applicability of environmental criminology to crimes in cyberspace, particularly evaluating the virtual places of cybercrime and how they differ to their physical counterparts. The lack of development of a 'cyberplace' concept is a critical gap in this area of research, and demonstrates the relative infancy of this new research direction.

In Chapter 6, I go much further than former studies in providing an overview of cybercrime research from two disciplinary perspectives: information security and environmental criminology. I draw parallels between these two understandings of cybercrime, highlighting areas of overlap, and reasoning that future works could utilise these complimentary approaches to cybercrime prevention in an holistic manner. I initiate this process, first, by demonstrating how techniques from environmental criminology could be applied to devise new cybercrime countermeasures (particularly for disrupting botnet and malware delivery operations), and second, by proposing a new concept of *cyberplace* and discussing how it may be applied in cybercrime analysis and prevention.

Following my work, other researchers have argued a similar standpoint of the need for an interdisciplinary approach to cybersecurity, but with a focus on addressing specific cybersecurity challenges. In one work [118], the authors argue the need for an holistic framework to understand and reduce human-related risks in cybersecurity and cybercrime ecosystems, drawing from a range of theoretical

concepts (technological advancements and social adoption, opportunity management, behavioural and business models). They report ongoing work in developing such an holistic, co-evolutionary framework for sociotechnical ecosystems, particularly with two use cases. In another study [59], the authors propose a framework for identifying unintended harms caused by cybersecurity countermeasures (such as criminal displacement or misuse). They argue that this framework could enable stakeholders in cyberphysical environments to implement countermeasures and risk management strategies with more thorough consideration. A similar work [159] considers how methods from opportunity reduction and behaviour change can be used to improve the precision of cybersecurity controls – precision that is purposed to protect legitimate users of sociotechnical systems, while simultaneously preventing malicious activities. These emerging works demonstrate increasing momentum and contributions of fields such as environmental criminology in the cybersecurity domain, especially regarding sociotechnical systems security.

### 2.7.2 Concepts of 'Place' in Cyberspace.

Researchers and professionals in a variety of fields have made concerted attempts to formally establish a concept of 'cyberplace', or virtual location.

In the geographical sciences, Tranos and Nijkamp [203] study the impact of physical distance on the formation of the Internet infrastructure, and whether physical distance survives in virtual geography, even after controlling for relational proximities. On the other hand, in the field of urban technology, Devriendt *et al*. [78] identify two approaches to analysing "virtual" or digital intercity linkages (i.e., linkages based on ICT). In both of these works, they utilise the same geographic metaphors of *cyber-place* (CP), which is defined as the projection of the infrastructural layer of cyberspace on traditional space, and *cyberspace* (CS), which is defined as the virtual or immaterial world wherein people communicate with each other via networked technologies, and that physical laws and aspects, such as distance and time, are practically irrelevant. My definition of cyberplace significantly differs to that of prior works in that it derives a holistic concept of cyberplace, which takes into account cyberspace, online activity and cybercrime, and their relationship with

the real world. Furthermore, much like how 'place' in the real word can be decomposed into three fundamental aspects [74], I define the three components of 'cyberplace' that encapsulates all of its characteristics. In a sense, my definition of cyberplace combines the CP and CS metaphors.

From a sociological perspective, Wellman [209, 210] characterises computer networks as social networks, and thus argues that they should not be studied in isolation, but as integral parts of everyday life. An example of such studies includes the work of Sussan *et al*. [195] on how cyberspace allows consumers to form virtual communities and engage in online word-of-mouth exchanges. Wellman [210] initially thinks of computer-to-computer interactions becoming increasingly "placeless". Nonetheless, in reference to the development of place-based social networks, the author refers to "online relationships and communities" being "truly in cyberplaces, and not just cyberspaces", potentially alluding to cyberplaces as online services that enable peer-to-peer networking.

Significant efforts have also been made in the legal sector in isolating a licit definition of 'place' in cyberspace. Hunter [113] discusses the CYBERSPACE AS PLACE legal metaphor, which was commonly used in the U.S. to understand Internet communication as "having certain spatial characteristics from our physical world experience", thus giving legal precedent in cases involving Internet services and digital property. Lemley [132] argues that the Internet is dominated by publicly accessible sites or spaces, therefore Law should not assume every part of cyberspace is "owned" by a particular entity. The author also contradicts the CYBERSPACE AS PLACE metaphor, mainly due to large disparities between the physical idea of property and the cyber world. However, the author does not discuss the use of synchronous applications, such as social networks or instant messengers (which connect people with similar interests much in the same way as place-based relationships), in relation to the CYBERSPACE AS PLACE metaphor. The author also fails to consider individual websites as places, which may serve as better 'place' analogies. Cohen and Hiller [69] discuss the legal definition of 'place' (U.S. Law) and attempt to define an analogous counterpart for 'cyberplace' for the purposes of clarifying

laws and rights surrounding such matters. In particular, the authors note that the CYBERSPACE AS PLACE (or Internet as a place) metaphor is far too broad a definition, and suggest a new framework that identifies when a private provider of online content or access creates a 'place of public communication'. The purpose of this framework is to disambiguate between private and public spaces on the Internet, much like in the physical world, for conflict resolution.

# Chapter 3

# Data Sources

In this chapter, I describe the data sources that are used to study the malicious file delivery ecosystem and specific delivery operations. I also describe the additional sources of ground truth that I use to enrich these data.

## 3.1 Symantec Download Metadata

I leverage a dataset shared through a research collaboration with Symantec. This gives me access to the fully anonymised data collected by its anti-virus and intrusion detection/prevention products on millions of end-hosts around the world. These datasets are collected from users who explicitly opt-in to the data sharing program, and does not include personally identifiable information (PII).

The dataset contains download activity information from real hosts for a period of one year between 1 October 2015 and 29 September 2016. The users that have explicitly opted into Symantec's data-sharing programme periodically transmit metadata on the binaries that they download. This dataset offers rich information regarding the time at which a binary is downloaded, from which server it is downloaded, and which program initialises the download activity. If a malicious file `4.exe` is downloaded from a website `http://avirivi.co.il/counter`, for example, the data will contain information about the file, as well as the website URL from which `4.exe` was downloaded, and the IP address of the server `198.252.64.124`. Note that if this malicious file `4.exe` downloaded other malicious files, I define `4.exe` as a *dropper*. Additionally, if a dropper malware sam-

ple downloads a second malicious file, the dataset will record information about both the server from which the file is downloaded and the dropper that initiated the download.

To be more precise, for each download event, the dataset contains the following information: the timestamp of the download event, the name, SHA-2 (256 bits) and size (in bytes) of the downloaded file, the host URL (with parameters omitted) and IP address of the server the file was downloaded from, the SHA-2 of the parent file which initiated the download, and the referral URL (with parameters omitted) that this program was originally referred from (if available).

I collect data on a daily basis in October 2015 (31 days) and, from then on, every Thursday on a weekly basis from November 2015 to September 2016 (47 days). In total, the dataset contains 129 million download events consisting of 21,398,564 unique binaries that are categorised as either PUP or malware. These binaries are downloaded from 12,394,454 unique URLs, hosted on 557,429 unique IPs. After IP filtering (see Section 4.2.1.1), these are reduced to 21,388,521 unique binaries, 12,390,735 unique URLs, and 553,812 unique IPs.

It is important to note that, although this dataset is several years old at the time of writing this thesis, the techniques derived using this dataset are timeless. Moreover, as the security community has found, malware operations often last for several years [128, 160]. As such, many malware that operate today were likely in operation at the time this data was collected. Finally, as I will show in the ensuing chapters, many malware behaviours are observed repeatedly throughout the literature. As such, the observations presented and lessons learned in this thesis will likely recur in modern-day malware, though some permutations may exist.

## 3.2 Ground Truth

I utilise a variety of ground truth data to enrich the Symantec download data. This is to establish whether files are malware or PUP, to what families they belong, and how their dropping networks evolve over time.

### 3.2.1 Symantec Reputation Scores

Symantec also employs extensive static and dynamic analysis systems to determine the maliciousness of a binary. My work focuses on malicious file downloads. To this end, this dataset is preprocessed to leverage the reputation score that Symantec associates to files, discard any file that has a high (benign) reputation score, and keep files that are involved in the delivery of malware or PUP (e.g., using the ground truth maintained by Symantec) or confirmed as malicious by VirusTotal [116]. These reputation scores also serve as additional ground truth for unlabelled files. Note that I consider a file to be 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. A similar technique has been used in other work [154, 194].

### 3.2.2 VirusTotal

VirusTotal [10] is a free online service that analyses submitted files and URLs across different antivirus engines and website scanners, aggregating these scan outputs. I query VirusTotal with each file SHA-2 to obtain the number of AV products that flag the file as malicious, as well as the AV-specific malware or PUP family labels designated to it.

**Ecosystem study.** For the measurement study in Chapter 4, I only collect VirusTotal data for download events occurring on October 1st, November 12th and 19th, and December 17th and 24th, 2015. This is because VirusTotal limits queries at a rate of 4 requests/minute for non-paying users. This throttling limited the amount of ground truth that could be collected for this study within a reasonable time period. Of course, however, workarounds to this limitation exists if adequate resources (time, funds) are available.

**Takedown study.** Coupled with throttling limitations, VirusTotal can sometimes take several months (or even years) to detect and classify some malicious files in the wild accurately [137, 130, 162]. As such, for the longitudinal study of malware delivery operations that faced takedown attempts in Chapter 5, I collect and analyse VirusTotal data for the remaining download events in 2015-16 approximately 3

years after first being observed (i.e., in 2019). This makes sense since, given the retrospective nature of this study, I seek to characterise the evolution of different malware and PUP operations as accurately as possible.

### 3.2.3 AVClass

In conjunction with VirusTotal, I utilise the AVClass malware labelling tool [177] to remove "noisy" and conflicting malware labels for a given sample 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 a single SHA-2 of the AdRotator PUP family. AVClass processes these VirusTotal labels to generate the `AdRotator` family label of the `PUP` software class for this same SHA-2. At times, a single family may be associated to file SHA-2s that are labelled as both PUP and malware. Therefore, I use majority voting on each family to label it and its associated SHA-2s as either PUP or malware.

**Ecosystem study.** I used the default AVClass family labels for the study in Chapter 4, given that the tool was developed around the same time the download data was observed.

**Takedown study.** I utilised an updated set of AVClass family labels[1] at the time of the study in Chapter 5.

### 3.2.4 National Software Reference Library

NSRL provides SHA-1 and SHA-2 hashes of known benign and reputable programs. I use NSRL's Reference Data Set (RDS) version 2.67 to identify benign files that are potentially involved in malicious file delivery.

In total, the ground truth dataset contains 1,034,763 malicious file SHA-2s (4.83% of all files), 443,541 (2.07%) of which is 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). Finally, the lack of ground

[1]Commit 21806f3 from `https://github.com/malicialab/avclass` (July 27th, 2018)

truth for the remaining 20,003,241 SHA-2s (93.5%) leave their relative benignity or malice unknown.

It is worth pointing out the issue of lack of ground truth is a common problem within the security community, mainly because of software polymorphism [31] and singleton binaries [138]. However, one must be clear that the primary focus of these studies is understanding the structures of *malicious* file delivery operations, how *known* operations respond to different mitigation strategies, and identifying pinch points within them. These studies *do not* aim to classify unlabelled files or solve the "ground truth problem."

## 3.3 Additional Data Sources

I enrich the dataset even further to establish ground truth on the network hosts that deliver the malicious files. This allows us to characterise upstream delivery networks with clarity and track how their use by malware and PUP operators evolve over time and in response to different mitigation strategies.

### 3.3.1 IP–ASN Mappings

I leverage a dataset of IP address to Autonomous System Number (ASN) mappings that was provided by Cambridge Computer Laboratory, University of Cambridge. This data was collected daily between October 1st, 2015 and September 29th, 2016. *I extend my gratitude to Dr. Richard Clayton for this dataset.*

### 3.3.2 Geolocation Data

To further characterise the locations of different delivery service providers (particularly in the takedown study in Chapter 5), I leverage a *geolocation* dataset to map IP addresses to the countries in which the servers are hosted. To achieve this, I use the `python-geoip` PyPI package[2], and MaxMind `GeoLite2` data[3] collected around the time of the study.

In particular, there are datasets that were collected on `20151027` and `20161203`. I found that for 98.3% of IPs in the Symantec dataset, the two Ge-

[2] `https://pypi.org/project/python-geoip/`
[3] `https://dev.maxmind.com/geoip/geoip2/geolite2/`

oLite2 datasets recorded the same geographic locations. This was unsurprising, as prior research has shown that most IP-geolocation structures are stable for static devices (servers, routers, desktops) [208]. Given the identical mappings between the two datasets, I opted to use the `20161203` GeoLite2 dataset for the study.

### 3.3.3 Mozilla Public Suffix List

The Public Suffix List is a cross-vendor initiative to provide an accurate list of domain suffixes. This list includes common CDN resources as suffixes (e.g., `ca-central-1.amazonaws.com`). I used the Mozilla Public Suffix List[4] to identify the effective second-level domains (e2LDs) in this dataset for clustering purposes. The list is editable, so I included `amazonaws.com` as a suffix to separate different users of its services (e.g., clients of Amazon AWS cloud services).

[4] `https://publicsuffix.org/`

# Chapter 4

# Measuring the Malicious File Delivery Ecosystem on the Web

In this chapter, I present my first experimental study: measuring the malicious file distribution ecosystem. This work was in collaboration with *Symantec Research Labs*, who collected and provided the download metadata from millions of their product users, as well as my doctoral supervisors, who both gave invaluable insights and direction over its *many* iterations. We published an ACM AsiaCCS conference paper based on this chapter titled, *'Waves of Malice: A Longitudinal Measurement of the Malicious File Delivery Ecosystem on the Web.'* The analysis code is publicly available on GitHub.[1]

## 4.1 Introduction

Malware delivery has become a major business in the cybercriminal economy. Through decades of evolution and refinement, cybercriminals have developed entire operations around delivering malicious payloads to end-users *at scale*, whether the payloads be proprietary (i.e., controlled by the same actor who delivers it) or third-party (i.e., controlled by a different actor to the one who delivers it).

There are myriad techniques cybercriminals use to deliver malware: transmission through physical media, social engineering (e.g., tricking a victim into downloading the malware from a malicious link or email attachment), drive-by down-

[1] `https://github.com/ColinIfe/mdn`

loads and exploit kits hosted on compromised websites, and malicious advertisements (or malvertisements).

Over time, the cybercriminal economy developed the *pay-per-install* (PPI) service model, which is characterised by cybercriminals paying for their malware to be installed onto end-user devices by the PPI network operator or by one of the operator's affiliate distributors. A core proponent of the PPI business model is the *dropper*, which is software designed specifically to download other software components onto victim devices.

As described in Section 1.1, researchers have recently uncovered a parallel economy of *potentially unwanted programs* (PUPs) [129, 127, 200], which share many traits with malware. Examples of this type of unwanted software include adware, spyware, and shady browser toolbars. Research has shown that PUP victims are usually tricked into installing a downloader, or *dropper*, through social engineering [127]. After such a dropper is installed, additional components are dropped through a PPI service [200].

Previous research has suggested that, although mostly disjoint, a consistent number of malicious actors (e.g., PPI operators) serve both malware and PUP samples. Kwon *et al*. [131] show that 36.7% of the droppers that they observed downloaded both malware and PUPs. Despite this finding, many questions remain unanswered on the structure, workings, and dynamics of malware delivery networks. What does the malicious file delivery network look like? Are there differences in the network structure of infrastructures that solely download malware, PUP, or both? How do these infrastructures change over time? Answers to such questions could help the security community better understand this malicious ecosystem, and could expose weak points in these infrastructures for takedowns.

In this study, I adopt a data-driven approach to provide a *longitudinal characterisation* of the malware and PUP delivery ecosystem on the Internet. First, I process 129 million download events collected from millions of real users who downloaded unwanted software over one year. This data contains information on the files downloaded, on the network servers that they were downloaded from, and

on the dropper file that initiated the download. I subsequently model these download relations as a graph and apply graph analysis techniques to identify the related network and file components. I then look at the types of files that these components download, and study their temporal behavioural characteristics over a short period (one day), as well as over a medium period (every day for a period of one month) and the long term (one day a week for a period of one year).

**Overview of results.** I show that the malicious file delivery landscape can be partitioned into two disjoint ecosystems: a tightly connected set of network infrastructures that are mostly responsible for downloading PUPs, and a set of isolated infrastructures that are mostly responsible for downloading malware. I also show that the PUP Ecosystem is stable over the long-term (i.e., one year). In raw numbers, the PUP Ecosystem is responsible for 80% of suspicious file downloads worldwide. Although previous research found that PUPs are pervasive in the wild [127], this work presents the first comparison of the prevalence of PUP and malware. I estimate the proportion of PUP-to-malware in the wild – roughly 5:1 in # of SHA-2s, and 17:2 in # of downloads – and analyse the characteristics and distribution patterns of their ecosystems. Confirming results from previous work [131], I show that these delivery infrastructures are often not responsible for delivering a single type of malicious file (i.e., PUP or malware), but, instead, often deliver both. I observe the activity patterns of distribution infrastructures over time and their lifespans. This study provides the security community with an unprecedented view of the characteristics of the malware and PUP delivery ecosystems. Also, I provide a methodology, with initial results, that identifies elements (IP addresses, autonomous systems, domain names) in a delivery infrastructure that do not change over time. These can be used to direct takedown efforts towards those elements that are not volatile and therefore could have an impact if taken down.

## 4.2 Methodology

This study leverages two stages of analysis: (i) a 24-hour snapshot analysis, and (ii) a longitudinal analysis. In the first stage, I group related hosts and files observed over 24 hours, and map the network infrastructures involved in the delivery of malicious files. In the second stage, I track the evolution and behaviours of these infrastructures. In this section, I describe these stages in detail.

### 4.2.1 Snapshot Analysis

The data processing pipeline for a 24-hour snapshot is as follows: i) IP filtering, ii) building the graph, iii) separating components, and iv) file classification.

#### 4.2.1.1 IP filtering

Since this dataset presents a global outlook on download data, files that appear to be generated from the host machine (localhost) or private IP addresses could be incorrectly inferred as being part of the same infrastructure. Consequently, IPv4/v6 addresses that are not valid for public use on the Internet [6] are removed. As a result, the graph-building stage ignores files/URLs that are *only* downloaded from / hosted on these IP addresses.

#### 4.2.1.2 Building the graph

I build a directed graph for each observation window (24 hours), defining a graph as

$$G = (V, E) \tag{4.1}$$

where $V$ is a set of heterogeneous nodes that represent the following entities: IP addresses, URLs, and files. $E$ is a set of directed edges that represent relationships between each node. Note that in this study, URLs that share a common fully qualified domain name (FQDN) are clustered together rather than explicitly defining them as nodes within the graph. It should be noted that, in a similar manner to the approaches used by other researchers [194, 24], I update this methodology in Chapter 5 to represent FQDNs as graph nodes explicitly, so as to identify URL-to-FQDN

relationships from the outset. For simplicity, I have outlined the additional steps in this chapter to avoid repetition later.

An example of a download graph is shown in Figure 4.1, which captures both the file dropping behaviours (client-side) and the upstream distribution network (server-side). To build the download graph, I take *download events* as the input, where each download event is represented as a tuple

$$\mathbf{d} =< I, D, U_r, ..., U_f, F_f, A_f, U_p, F_p > \tag{4.2}$$

where $I$ is the IP address from which the file was downloaded, $D$ is its FQDN (Chapter 5 only), $U_f$ is the host URL of the download (after removing the URL parameters), while $F_f$ 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 the "reputation" and "prevalence" scores assigned to these files by Symantec's static and dynamic analysis systems (see Section 3.2.1). Information on any HTTP redirection chains that are involved in the download event is also included, terminating with the download URL $U_f$. If the download event takes advantage of redirection, this initiating referrer URL, $U_r$, would be recorded. I use this information because, as previous research has shown, malware operators utilise redirections to make their infrastructures more resilient and difficult to detect [193]. Finally, information about the parent file that initiated the download event is retained, which may be as a result of a user-triggered download from a legitimate program, such as a browser or an installer, or a malware sample dropping other malware, such as through a pay-per-install scheme [47]. $F_p$ represents this parent file as identified by its SHA-2, whereas $U_p$ indicates the URL from which this parent file was downloaded.

The following steps are then taken to build the graph with a download event **d**:

- For each element $d_e$ in the tuple **d**, I check if $d_e$ already has a node in the graph. If it does not, I create a new node with a unique identifier and add it to the set of nodes $V$. I use the full IP address as an identifier for IP nodes, the

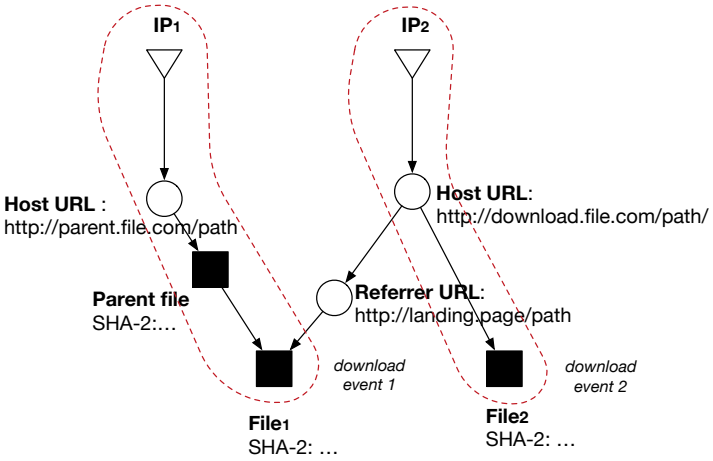


**Figure 4.1:** An example of a download graph with two series of download events highlighted. This schema is used for this study, while an updated schema is adopted for a later study in Section 5.3.1.

FQDN for FQDN nodes (Chapter 5 only), the full URL without parameters for URL nodes, and the SHA-2 hash for file nodes.

- If there is no pre-existing edge between any two elements $d_{e1}$ and $d_{e2}$ in $\mathbf{d}$, I create an edge with weight 1 and add it to the set of edges $E$. If the edge already exists, its weight is incremented by 1. The following directed edge relationships are permitted between each node type (as represented by each element $d_e$), subject to their presence in the download event $\mathbf{d}$:
  - FQDN $d_D \rightarrow$ URL $d_U$ (Chapter 5 only),
  - download IP $d_I \rightarrow$ download URL $d_{Uf}$,
  - download URL $d_{Uf} \rightarrow$ referrer URL $d_{Ur}$,
  - in decreasing order of precedence, subject to presence in the download event: (i) referrer URL $d_{Ur}$, (ii) download URL $d_{Uf}$, or (iii) download IP $d_I \rightarrow$ downloaded file $d_{Ff}$ (or parent file $d_{Fp}$),
  - parent file (dropper) $d_{Fp} \rightarrow$ downloaded file $d_{Ff}$.

Take *download event 1* in Figure 4.1, for example: $File_1$ is dropped by *Parent file*, which was downloaded from host URL `http://parent.file.com/path`, hosted on IP address $IP_1$. In *download event 2*, $File_2$ was downloaded from host

URL `http://download.file.com/path/` hosted on $IP_2$. These two disconnected graphs are connected by the third download event where $File_1$ was downloaded via referrer URL `http://landing.page/path` leading to host URL `http://download.file.com/path/`.

#### 4.2.1.3 Separating components

The primary step towards attributing files, hosts, and their activities to actors is to separate the directed download graph into *weakly (undirected) connected components*. This enables me to identify distribution networks of files and hosts that have direct interactions with each other, and characterise them as independent structures for a given 24-hour period. The graph structure is divided into file-only and network-only (sub)components, which are the connected components derived from the file-only and network-only sub-graphs. I define (i) a *network infrastructure* as a component in the network-only subgraph, while in the case of a file-only subgraph, (ii) a *file infrastructure* as a component consisting of *at least* two file nodes, and (iii) a *lone file* as an isolated node in this subgraph. For example, Figure 4.1 shows two network infrastructures, $\{\text{IP}_1, \text{HostURL}\}$ and $\{\text{IP}_2, \text{HostURL}, \text{ReferrerURL}\}$, one file infrastructure, $\{\text{ParentFile}, \text{File}_1\}$, and a lone file, $\{\text{File}_2\}$. This separation into sub-graphs assists in the task of attributing infrastructures to independent actors and tracking these over time.

#### 4.2.1.4 File classification

To further understand the malicious file delivery ecosystem, I am interested in labeling graph components as "malware," "PUP," or "unclassified," based on the most common types of files of which they consist. VirusTotal [10] is a freely accessible site that analyses file submissions across dozens of antivirus engines and produces detailed reports and detection statistics. Amongst these statistics are the family labels by which each antivirus engine classifies the file (e.g., a prominent malware or PUP family).

Simple majority voting could be applied to all labels produced in a VirusTotal report. However, an issue with this approach is that antivirus vendors use inconsistent labels for positive samples, even when the same malware families are de-

tected. For example, two engines may generate labels of `Adware.Rotator.F` and `Adware.Adrotator.Gen!Pac` for the same instance of the AdRotator PUP family. These inconsistencies lead to unreliable majority votes. As a result, Sebastian *et al*. [177] designed and evaluated the AVClass malware labeling tool to overcome this problem.

In this study, the AVClass tool is used to label each file SHA-2 that generates a VirusTotal response with a family name, and as likely malware or PUP. Each graph component is then assigned a malware, PUP, or unclassified label, based on a majority vote on the most common family it distributes. If VirusTotal classifies a sample as malicious, but AVClass does not contain its label in its database of aliases, I label it as a singleton cluster named after its SHA-2.

### 4.2.2 Longitudinal Analysis

After mapping the actors involved in malicious file delivery over one day, we want to understand how stable these distribution infrastructures are over time. To this end, file-only and network-only components are tracked on a daily and weekly basis (working from the same day of the week) over an entire year. I also track the lifespans of these infrastructures over a year, using a weekly sampling frame, with respect to the first day of the dataset. More precisely, I do the following:

#### 4.2.2.1 Snapshot processing

For each day of data, I generate file-only and network-only connected components. To achieve this, I repeat the steps from *Snapshot Analysis* in Section 4.2.1 to build components from the overall graphs. I also generate file-only and network-only sub-graphs and build components from these.

#### 4.2.2.2 Optimal signature selection

To track distribution infrastructures across different days, I need to first characterise each graph component with a *signature*: a set of nodes within these components which are likely to be temporally stable. Therefore, I need to determine (i) a good criterion for node stability, and (ii) a suitable signature length. The following experiments are conducted to establish suitable signature characteristics:

(1) **Node centralities.** I pursue a suitable criterion for identifying stable nodes through graph percolation, i.e., the breaking down of a graph component by systematically removing nodes. Graph percolation [49] is useful in showing how resilient a network is to disruption, and by what method. I utilise different *node centralities* as the criteria for selecting the node to be removed at each iteration, with the idea that stable 'root' and 'branch' nodes (e.g., IP addresses, hosts, droppers) are likely to be more "influential" than ephemeral 'leaf' nodes (i.e., end-user downloads). In this case, I use node centralities as proxies for "influence." I then conduct graph percolation on a graph component, via centrality criteria, until it completely disintegrates. I compare the rates of graph percolation under different node centralities and select the one with the highest rate.

(2) **Sensitivity analysis.** Besides identifying the ranking metric of the nodes most likely to be stable, I also need to determine a suitable number of nodes to include in the tracking signature when I attempt to trace infrastructures across days. Intuitively, it is unlikely that I would need to consider every single node in a given infrastructure in this matching process. To this end, I conduct a sensitivity analysis using the node selection criterion as well as a range of signature lengths as I measure the number of infrastructures that I can track across a pair of days. I then select a maximum signature length based on the principle of diminishing returns, i.e., when the increase in tracking accuracy is insignificant in comparison to the increase in signature length. I present the results of these experiments in Section 4.3.2.2.

#### 4.2.2.3 Component tracking

I have defined how I generate the signature of each component. Now, I explain how I track these in time.

For any pair of consecutive days, i.e., day $i$ and day $i+7$, I generate a bipartite graph: a vertex set $V_i$, representing components from day $i$, and a vertex set $V_{i+7}$, representing components from day $i+7$. Each component is represented as a single vertex, $v$, with an associated component signature, $s$. For example, component $v_{i,j}$ represents the $j$th component from day $i$ and has signature $s_{i,j}$.

Edges represent matches between component signatures when their intersection is a non-empty set (i.e., $s_{i,j} \cap s_{i+7,k} \neq \emptyset$). This representation enables us to generate a simplified, one-to-one mapping of matched components via the following rules (in order of priority):

1. If $v_{i,j}$ and $v_{i+7,k}$ share an edge, and they have no other incident edges, I retain this edge as a *simple transition*.

2. If $v_{i,j}$ shares edges with multiple vertices from $V_{i+7}$, the "best match" is chosen (see below).

3. If $v_{i+7,k}$ shares edges with multiple vertices from $V_i$, the "best match" is chosen.

The "best match" algorithm works as follows:

1. Retain edge with the smallest difference in component size.

2. If multiple edges retained, retain edge with the greatest overlap of leaf nodes between components (i.e., the same payloads).

3. If multiple edges retained still, retain one of the edges by random.

Forward-facing transitions are prioritised over backward-facing ones, trading-off a little tracking reliability for simplicity. The "best match" algorithm assumes that there is more stability in how many files a dropper distributes over which files it distributes. This assumption is supported by the observation that malware can undergo rapid polymorphism [31]. Note that this tracking technique is also limited in that it oversimplifies the splitting or joining of infrastructures across days as straightforward transitions. Nonetheless, this is sufficient in estimating the activities and lifespans of these delivery infrastructures, giving lower bounds for such.

In Section 4.3.2, I provide the longitudinal analysis of the data, particularly focusing on the *retention rates* of components over time. This aspect is indeed interesting to understand how ephemeral malicious file operations are and to better understand which mitigation techniques are more promising against these phenomena.

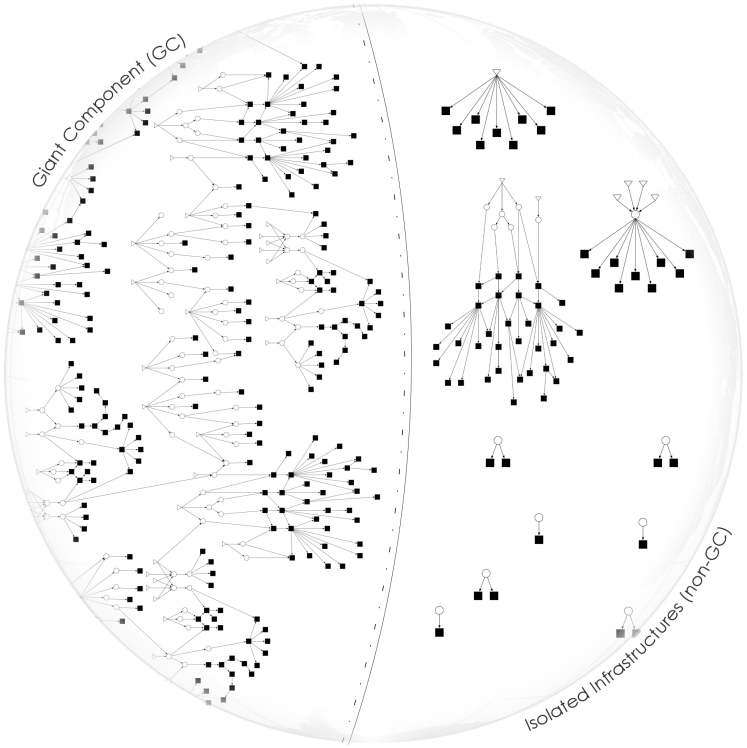


**Figure 4.2:** Illustration of file distribution infrastructures. White triangles represent IP addresses; white circles download and redirection URLs; and black squares files.

## 4.3 Analysis

As explained in the previous section, the analysis is in two stages: first, I look at a single day of data, to better understand the network and file infrastructures involved in the malicious file delivery landscape. I then look at multiple days, to see how the network and file infrastructures evolve over time. In this section, I illustrate the results of this analysis in detail.

### 4.3.1 Snapshot Analysis

I build the graph for the first day of the collection period, 1st October 2015. After the pre-filtering operations described in Section 4.2.1.1, I obtain a graph $G$ with 1,661,636 nodes and 1,930,648 edges. These nodes consist of 964,998 file nodes (SHA-2s), 385,861 host URL and 218,530 referrer URL nodes (130,630 domains), and 92,247 IP nodes. Each file node represents all download events relating to a unique file, identified by its SHA-2, with a total of 1,644,906 download events

**Table 4.1:** Top 10 countries by # of GC articulation IP nodes.

| Region | Art. IP nodes | Region | Art. IP nodes |
|---|---|---|---|
| United States | 1419 | Russian Federation | 39 |
| China | 268 | Canada | 31 |
| Netherlands | 147 | United Kingdom | 31 |
| France | 114 | Luxembourg | 28 |
| Germany | 53 | Brazil | 26 |

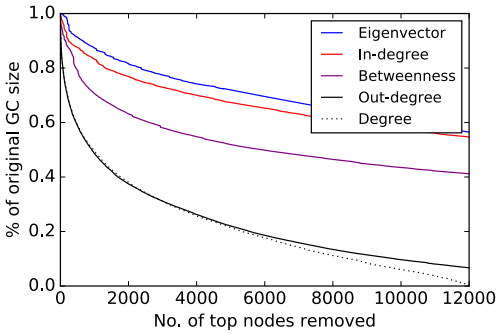


**Figure 4.3:** Decay of the GC by graph percolation under different selection criteria. N.B. line order follows graph legend.

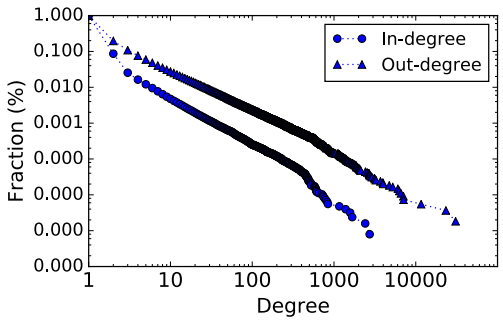


**Figure 4.4:** Giant Component degree distribution (complementary cumulative distribution function).

recorded for the first day. The graph is separated into weakly connected components (see Section 4.2.1.2). Consequently, 58,173 connected components are generated.

I find that a Giant Component (GC) emerges, which accounts for 80% of download activity, comprising of 786,240 unique files (1,345,586 download events) distributed through 89,550 domains, 480,110 URLs, and 51,436 IP addresses. The GC comprises network components and file components interconnected with each other, such as multiple network infrastructures dropping the same set of files. To put this into perspective, the next largest non-Giant component consists of only 2,000 nodes. The remainder of download activity (which I refer to as the Non-Giant Component or NGC) is attributed to 58,172 independent distribution infrastructures. Figure 4.2 shows an illustration of the two emergent download ecosystems.

#### 4.3.1.1 Graph structural characteristics

It is pertinent to verify whether the GC identified is indeed a well-connected set of network infrastructures, or if it is an artifact of the methodology. To this end,

I conduct graph percolation as described in Section 4.2.2.2, shown in Figure 4.3. I find that the GC is tightly connected to a minority of nodes. For instance, it is required to remove over 1k (0.08%) of the highest degree nodes to reduce the size of the GC by more than 50%, and at least 6k (0.46%) nodes – 5.5k of which are network nodes – to reduce the size by 80%. This ratio is an extreme example of the Pareto principle, which itself states that for many real-world outcomes, roughly 80% of effects come from 20% of causes.

Following from the graph percolation experiment, I identify the articulation nodes which form the structural backbone of the GC. Table 4.1 shows that, when I focus on IP addresses, the United States is the biggest regional contributor to this massive distribution infrastructure. This ranking could indicate where ISP take-down efforts would be most effective in dealing with unwanted software distribution, notwithstanding the potentially disproportionate number of ISPs located in the US. The GC is an approximate *scale-free network*: Figure 4.4 shows its degree distribution approximately following a power-law distribution. It contains 1.3M nodes and 1.6M edges. The diameter of the GC is 20, meaning that there are only 20 hops along the longest chain of IPs, URLs, and dropped files. The average path length of the GC is 6.20 (average number of hops between any pair of nodes). The GC also has a global clustering coefficient of $3.6 \times 10^{-5}$. This property could be an indication of a tree-like structure for the GC, with a relatively small number of highly interconnected root nodes, but many branches and leaf nodes. This conclusion is supported by the fact that only a very small proportion of nodes – most of which are hosts – are responsible for the connectivity of most of the GC.

#### 4.3.1.2 Significance of the GC

Though initial findings showed that the GC is a well-connected ecosystem of files and network infrastructures, this component could still be an artifact of, for example, the shared use of IP addresses by different malicious operations due to their use of popular hosting providers and content distribution networks (CDNs). This classification would result in a false connection of services that are effectively independent in the real world, e.g., separately owned Amazon EC2 instances being

linked to the same IPs and/or domains. To rule out these scenarios, I conduct two experiments: first, I rebuild the graph without IP addresses, and second, I blacklist the most popular effective second-level domains (e2LDs). I use the Mozilla Public Suffix List[2] to identify the e2LDs in this dataset. Note that this list includes common CDN resources as suffixes (e.g., `ca-central-1.amazonaws.com`). I included `amazonaws.com` as a suffix to separate users of its services.

For these verification experiments, the graph $G$ is rebuilt without any IP address nodes, or any URLs with IP addresses in place of domain names (e.g., `http://119.147.227.164/path/to/file`). This results in a graph of 1,544,062 nodes (7% reduction) and 1,578,585 edges (18% reduction). After computing the weakly connected graph components, I find that the GC is considerably smaller, but remains stable, with 908,029 nodes (31% reduction) and 1,102,300 edges (32% reduction). Evidently, IP addresses help form a significant part of the GC, connecting about 31% of this component. In real terms, shared IPs connect a significant proportion of the unwanted software distribution market – potentially an indication of shared or repeated use of network infrastructure, or these services being illicit, thus not appearing on the public suffix list. However, these results show that there is also a strong interconnection between distribution services through URL-to-URL redirections between hosts, the distribution of multiple software per service (one-to-many) and and the distribution of common software between multiple services (many-to-one).

Next, the most popular e2LDs are categorised by grouping the network nodes (hosts and referrals) that share the same e2LD and rank them by the number of associated network nodes. Table 4.2 shows the top 20 e2LDs. I find that the top GC domains predominantly belong to popular CDNs, such as *MediaFire* (7.4k nodes), *Windows Azure* (under `msecnd.net`, 6.4k nodes), *Softonic* (2.7k nodes), and *Google* (2k nodes). An apparent *zz-download-zz* CDN is also very prominent, consisting of 2.86% (7.6k nodes) of hosts. As later results suggest, the unwanted

[2]The Public Suffix List is a cross-vendor initiative to provide an accurate list of domain suffixes – `https://publicsuffix.org/`

| Rank | e2LD | % of hosts | Rank | e2LD | % of hosts |
|---|---|---|---|---|---|
| 1 | mediafire.com | 2.80% | 11 | d3s8yh4ki1ad1i.cloudfront.net | 0.67% |
| 2 | msecnd.net | 2.40% | 12 | drp.su | 0.64% |
| 3 | uploaded.net | 1.70% | 13 | crusharcade.com | 0.62% |
| 4 | magnodnw.com | 1.56% | 14 | doff.info | 0.58% |
| 5 | mysimplefile.com | 1.03% | 15 | 4shared.com | 0.53% |
| 6 | softonic.com | 1.00% | 16 | zz-download-zz8.com | 0.51% |
| 7 | clipconverter.cc | 0.84% | 17 | zz-download-zz10.com | 0.50% |
| 8 | google.com | 0.77% | 18 | zz-download-zz7.com | 0.49% |
| 9 | file8desktop.com | 0.73% | 19 | mountspace.com | 0.47% |
| 10 | up1004.info | 0.72% | 20 | zz-download-zz9.com | 0.48% |

**Table 4.2:** Top second-level domains ranked by # of GC network nodes.

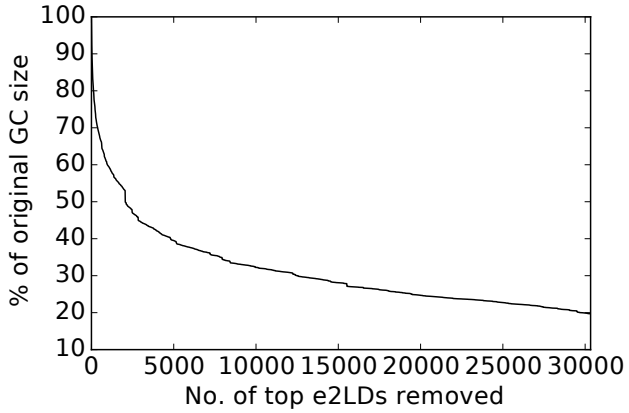


**Figure 4.5:** Decay of the GC (no IPs) by removal of top e2LDs.

software distribution economy may be leveraging, if not directly using, the infrastructures of benign and popular CDNs.

Figure 4.5 exhibits the exponential decay of the GC as the top e2LDs are removed in order of decreasing rank. I find that the GC structure remains resilient to percolation, even after the total removal of all 30,330 e2LDs in the GC. This result shows that both IP addresses and popular domains are important for the connectivity of the GC, but there is still a strong and resilient interconnectivity between the files that are distributed within it, as evidenced by 20% of the GC (180k nodes) remaining after removing all domains and IPs. That is, droppers also contribute significantly to the proliferation of unwanted software and are core to the malicious file delivery ecosystem. Taking into account that the smallest estimate of the GC is still *90 times* the size of the largest non-GC infrastructure (2k nodes), this evidence strongly suggests the real-world presence of the GC structure in the malicious soft-

ware delivery ecosystem, regardless of potential measurement artifacts. Moving forward, I proceed to study the differences between the GC and NGC infrastructures.

#### 4.3.1.3 File distribution of the GC and NGC

After identifying the presence of two distinct groups of infrastructures, the GC and the NGC (composed of 58k independent infrastructures), I aim to better understand what kinds of files are installed as part of the two ecosystems. The file classification process as described in Section 4.2.1.4 is applied. Of the 965k unique files downloaded in this day of data, VirusTotal generates analysis reports for only 80k file SHA-2s. AVClass [177] then processes the VirusTotal results to produce 61k family labels: 42k are from known families, while 19k are from unclassified families, which are labeled as "singletons" (see File Classification in Section 4.2.1.4). The remaining 19k of SHA-2s analysed by VirusTotal had not been classified as malicious at the time the download metadata was collected.

The attrition in ground-truth data is undesirable but expected. Only a small proportion of files are actually submitted to VirusTotal for analysis, hence the considerably small record size compared to the total number of files. Of the files for which VirusTotal has analysis records, some attain no AV detections, hence leading to AVClass producing no family labels for these SHA-2s. Even for files that have been detected as potentially malicious, some of them are only given generic labels by the AV vendors that detect them (e.g., `Trojan.Dropper.Gen`). These generic labels are stripped away by AVClass, leaving only family-specific labels, or, when none such labels exist, singleton SHA-2 labels for unclassified families. Because of this attrition in ground-truth, I instead use the available AVClass labels to characterise clusters of files that exhibit dropping behaviours. In particular, I use a majority voting scheme to label each file-only graph component with its most common family, as well as whether it is likely 'malware,' 'PUP,' or 'unclassified.' This estimation helps to characterise the remaining unlabelled but related files.

Figure 4.6 shows various family distributions for the GC and NGC ecosystems. A key finding here is that there is a clear difference in the presence of unwanted

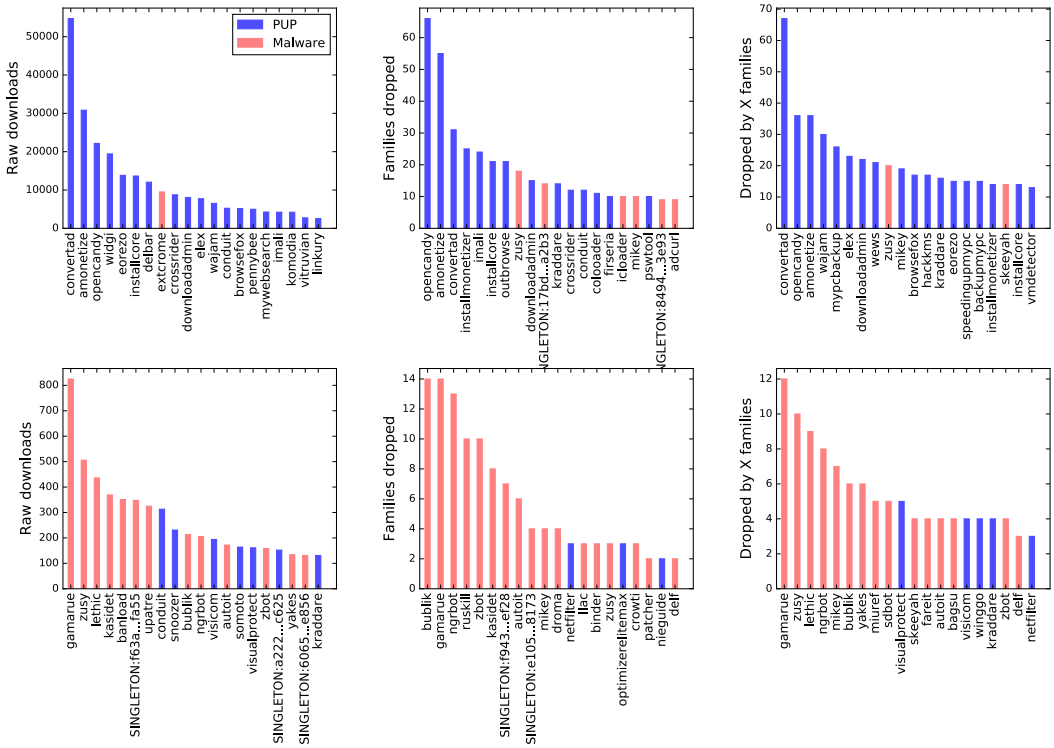


**Figure 4.6:** Malware/PUP family distributions. From left to right, figures show: i) top families by # of raw downloads; ii) top droppers by # of known families dropped; and iii) top known families dropped. The top row is for the Giant Component, while the bottom row is for Non-Giant Components.

software within these two ecosystems: the GC is primarily dominated by PUP, while the NGC is dominated by malware distribution activities. For the GC, PUP such as `convertad`, `amonetize`, and `opencandy` conduct the lion's share of download activity. Similarly, these families act as prominent droppers, installing other malicious files on infected computers, though there is also a considerable malware presence, particularly with the `zusy` family. In the NGC infrastructures, `gamarue`, for example, is very prolfic in both downloads and dropping activities, as are other malware families. It is worth nothing that `extcrome` is labeled as malware, while this family is actually adware and should, therefore, be classified as PUP [1]. This is a false positive result of AVClass, highlighting the imperfection of the AVClass labeller, although such misclassifications are generally rare in this dataset.

Another interesting observation is in the mixed presence of PUP and malware droppers and payloads within the GC. Given that the GC is a single, networked download infrastructure, this alludes to a mixed distribution mechanism for PUP and malware, although it is still PUP-dominated. By majority voting on the most common family for a given file component (see Section 4.2.1.4), I estimate that the numbers of independent PUP and malware file delivery operations (i.e., file components) in the GC are roughly 1.5k and 360, respectively (3.2k unclassified), and for the NGC, 190 and 250, respectively (2.9k unclassified). Note that I do not consider lone files as file delivery operations (i.e., singleton file components that do not engage in any dropping activities). 82 (1% of) file delivery operations involve both PUP and malware, which is in alignment with the findings of Kotzias *et al*. [127] that refer to PUP distribution and malware distribution being largely disjoint. However, I find that a single, massive file delivery operation that is a subset of the GC involves both PUP and malware, and is responsible for the distribution of 61k SHA-2s (7.7% of the GC) and 394k raw downloads (29% of the GC). This is in line with the work by Kwon *et al*. [130], who found that 36.7% of the droppers that they observed were downloading both malware and PUPs. To provide context, the next largest delivery infrastructure in the dataset only distributes 2k SHA-2s.

I also compute estimates of the proportions of PUP-to-malware in the wild by identifying SHA-2s of known families, and whether they are likely malware or PUP. In the overall graph $G$, the PUP-to-malware ratios are roughly 5:1 (SHA-2s) and 17:2 (raw downloads). The proportions of PUP-to-malware in the GC are roughly 8:1 in # of SHA-2s and 11:1 in # of raw downloads . In the NGC, the PUP-to-malware ratios are 1:1.78 in # of SHA-2s and 1:2.15 in # of raw downloads. Despite previous work already highlighting that PUP is more predominant in the wild than previously thought [127], this study was the first to quantify the ratio of malware and PUP in the wild.

#### 4.3.1.4 Case study: Opencandy operation

Figure 4.7 shows the known families dropped by the prevalent `opencandy` PUP, a commercial and popular pay-per-install (PPI) software, in this 24-hour snapshot.

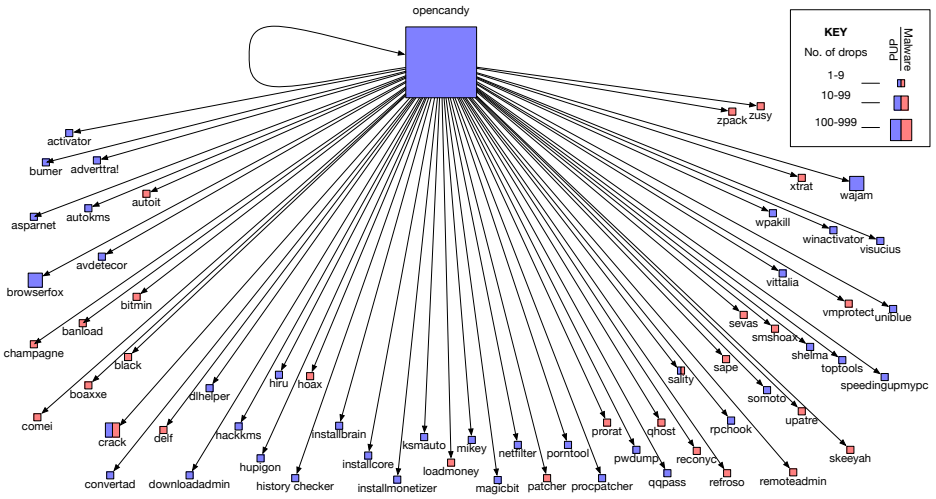


**Figure 4.7:** Known families dropped by Opencandy. Note that unknown families are omitted from this diagram.

This figure indicates that PUP-malware relationships and mixed distribution infrastructures may be a bigger problem than first thought. Opencandy seems to drop malware and PUP by similar proportions: 26 malware families (63 file SHA-2s) versus 37 PUP families (132 file SHA-2s), excluding the 288 Opencandy self-dropped SHA-2s. It is also interesting to see the dropping behaviours of this PPI. In particular, some of its customers include other installer software such as `convertad` and `installmonetizer`. This could be evidence of business-to-business relationships and shared distribution infrastructures between these competing PPI brands.

Opencandy also directly drops instances of its own binaries. I find that the longest chain of Opencandy dropping its own binaries is a length of 2 sequential drops. For instance, a drop-chain of Opencandy binaries (same SHA-2) have the file names `PowerISO5 X64.exe`, `ADV_35.EXE`, and `spstub[1].exe`. `PowerISO5 X64.exe` is the brand name of a CD/DVD image processing tool, while `spstub[1].exe` is the name of software developed by Conduit, most often with the description "Search Protect by Conduit". This could simply be the result of affiliate tracking. However, given that Opencandy has been found to distribute malware, one cannot completely rule out the possibility of foul play in the use of this mechanism.

Summarising this section, I discovered two file delivery ecosystems. The GC consists of interconnected file and network infrastructures and mostly drops PUP, while the NGC is composed of independent components and mostly drops malware. Because of the GC predominantly dropping PUP and the NGC mostly being responsible for malware downloads, for the remainder of this chapter, I can refer to the GC as the *PUP Ecosystem* and the NGC as the *Malware Ecosystem*.

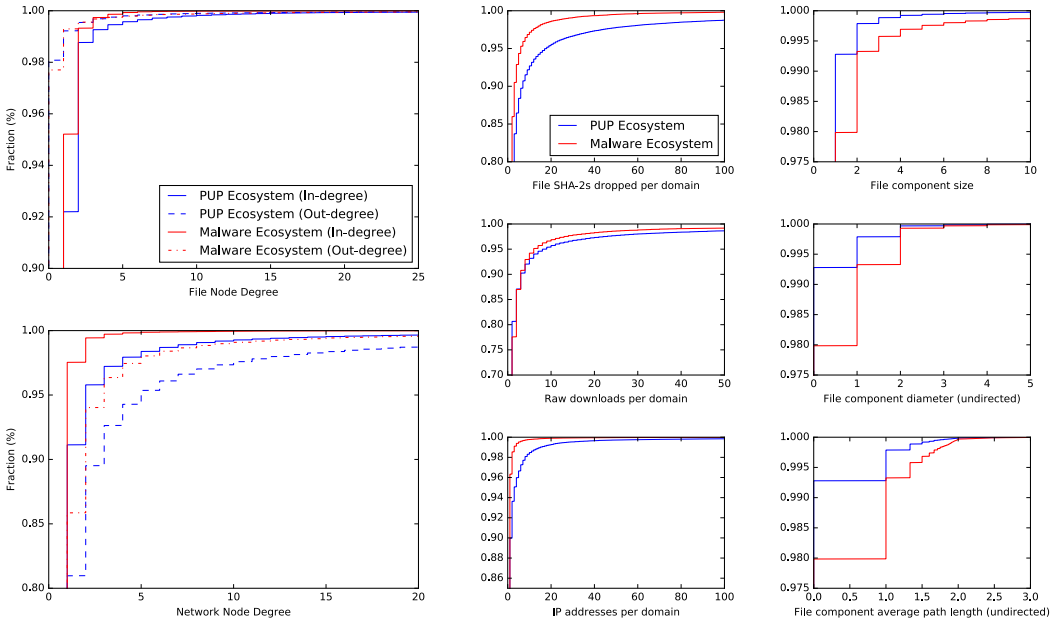


**Figure 4.8:** Structural comparison of PUP and Malware Ecosystems.

#### 4.3.1.5 Network and file characteristics of the two ecosystems

Figure 4.8 shows a structural comparison of the PUP Ecosystem and Malware Ecosystem sub-graphs. The file in-degree and out-degree distributions for the PUP and Malware Ecosystems are very similar. This could be indicative of largely similar distribution patterns being employed by malware and PUP authors, e.g., the common use of PPI services. However, the PUP Ecosystem generally has higher in-degree and out-degree distributions for network nodes. This result suggests several notions. First, hosts in the PUP Ecosystem are typically more interconnected (i.e., redirections between hosts) and/or utilise more IPs than hosts in the Malware Ecosystem. Also, hosts in the PUP Ecosystem are likely to be more prolific distrib-

utors (e.g., CDNs) than in the Malware Ecosystem, as also shown in the long-tails. This is likely due to the larger volume of traffic that these services can attract.

The file SHA-2s dropped per domain distribution shows that domains in the Malware Ecosystem download significantly fewer unique files onto victim systems than those in the PUP Ecosystem. However, the actual number of raw files downloaded by PUP domains is only slightly more. There are several possible explanations for this. Sites hosting malware could be used by malware authors to only distribute their own binaries, or by illegitimate PPI infrastructures that serve fewer malware customers per domain. The malware sites could also be distributing a few file SHA-2s before changing domain names in order to evade detection. On the other hand, while many of the sites in the PUP Ecosystem may be CDNs that are accessed explicitly by users to download different types of software (hence its larger distribution of SHA-2s), more of the malware-hosting sites could be benign sites that are compromised and unknowingly hosting exploit kits. In this case, victims would be infected without consent through silent drive-by downloads (hence the fewer SHA-2s distributed by Malware Ecosystem domains).

Over 98% of SHA-2s are *lone* files, as shown by the file component distributions. Lone files do not engage in any file dropping activities, nor are they dropped by any other file SHA-2 – they are observed to be downloaded only directly from hosts. Though component sizes vary, a majority of file components in both the PUP and Malware Ecosystems have diameters and average path lengths between 0 and 2 ($>$99.9% for both), although the file component sizes, diameters, and average path lengths in the Malware Ecosystem are slightly larger in general. This explains the very low clustering coefficient of the PUP Ecosystem (GC) and supports the notion that downloader graphs are generally very sparse and tree-like, with the Malware Ecosystem having similar, albeit unconnected, distribution patterns.

#### 4.3.1.6 Evasion tactics

The distribution of IP addresses per domain provides an interesting result. While there is evidence of over 90% of domains having only one IP address each, far more IPs per domain are used by a significant proportion of the PUP Ecosystem than the

| AS No. | Organisation | Region | Network Infrastructures Hosted |
|---|---|---|---|
| 16509 | Amazon.com Inc. | US | 2901 |
| 15169 | Google Inc. | US | 2508 |
| 14618 | Amazon.com Inc. | US | 1425 |
| 16276 | OVH SAS | FR | 1289 |
| 4134 | China Telecom | CN | 999 |
| 13335 | CloudFlare Inc. | US | 788 |
| 20940 | Akamai Technologies | EU | 755 |
| 24940 | Hetzner Online | DE | 600 |
| 4837 | China Unicom | CN | 567 |
| 26496 | GoDaddy.com LLC | US | 563 |

**Table 4.3:** Top 10 autonomous systems by # of network infrastructures hosted (i.e., connected components from network-only graph).

Malware Ecosystem. The high usage of IPs per domain in the PUP Ecosystem could be evidence of increased use of the fast flux technique in this ecosystem. However, this could also be attributable to the significant presence of various CDNs in this ecosystem, which has already been confirmed in previous sections.

Rossow *et al.* [172] state that rather than using servers with fast flux, some pay-per-install operators opt to distribute their dropper malware through multiple servers, each hosted on a different autonomous system (AS). Figure 4.9 shows the distributions of IPs and ASes being used to serve droppers. As I will show, there are fundamental differences in the dropping modus operandi between large portions of the PUP and Malware Ecosystems. While only less than 10% of droppers in the Malware Ecosystem are distributed across more than one IP or AS, over 70% of droppers in the PUP Ecosystem are distributed across more than one IP address, while over 45% are distributed across more than one AS, indicating the use of the aforementioned distribution tactic [172]. Servers with this abundance of resources are very likely to be constituents of CDNs. In fact, many of these malicious network infrastructures appear to congregate on well-known ASes, as shown in Table 4.3. Note that a single network distribution infrastructure may operate across multiple ASes.

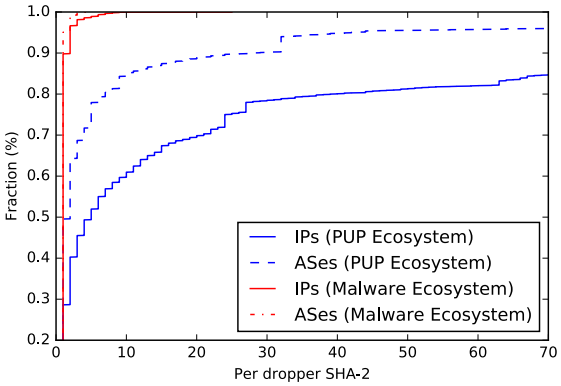


**Figure 4.9:** Distribution of IP addresses/autonomous systems serving each dropper. Droppers with no traceable IPs or ASes are omitted.

#### 4.3.1.7 PPI estimation

I also estimate the number of Pay-per-Install (PPI) services active during this single day. I define a PPI service as a network-only component that directly drops more than one type of malware or PUP family. I only consider known families, as the families of files with singleton AVClass labels could not be determined. I also aggregate network components with common e2LDs as they would represent common services. As a result, I estimate a potential lower bound of 215 PPIs operating in the PUP Ecosystem and 179 PPIs operating in the Malware Ecosystem. I note that the largest "PPIs" in the PUP Ecosystem and Malware Ecosystem involve about 99% and 24% of all e2LDs and IP addresses in their respective ecosystems. In real terms, this could further indicate that PPIs, as we know them, are more highly connected than once thought, either through shared use of infrastructure or from one service reselling to another. Note that other inter-host relationships (such as web links between pages) are not considered.

Finding such a high level of connectivity once again raises the question: why does the GC exist? Other works suggest that many different companies engage in unwanted software distribution and that it is unlikely that they are in close collaboration. However, arguably, this data suggests otherwise. It is possible that different affiliates are distributing the same binaries, or that software authors are running the same auction systems, leading to the downloads of these

same binaries. For instance, at least 4% of file SHA-2s in the GC are distributed by more than one host. Alternative hypotheses are that multiple companies that distribute unwanted software are actually controlled by a single company and/or many CDNs are acting as resellers unto other resellers, and so on. For example, a particular network infrastructure consists of 2 IPs and 30 different e2LDs, including `downloadopencloud.com`, `opencloudsafe.com`, `setupfreesoftware.com`, and `thesafedownload.com`, and, within the data, most major CDNs are structurally connected in one way or the other. Nonetheless, we can confidently rule out malware delivery mostly being a set of vertically integrated operations. Instead, it is either one big organised operation, or, perhaps more likely, a well-connected marketplace of infrastructure providers.

#### 4.3.1.8 Summary of results

In this 24-hour snapshot analysis, I showed that the malicious file delivery landscape could be partitioned into two disjoint ecosystems: a tightly connected ecosystem that is mostly responsible for downloading PUP, and a set of isolated infrastructures that are mostly responsible for downloading malware. I showed that the PUP Ecosystem is responsible for 80% of the total number of suspicious file downloads worldwide. I reckon that it is likely a well-connected marketplace of infrastructure providers. I calculated the ratio of malware and PUP appearing in the wild, and showed that PUP dominates malware by a ratio of 17:2 in the number of files downloaded worldwide. I compared the structures and distribution techniques of the two ecosystems, showing that PUP operators are more likely to distribute the delivery of their malicious files across more IP addresses and autonomous systems. I also showed that IPs from the U.S. are core to the PUP Ecosystem, which could be the most effective target for ISP takedowns. Using this technique, one could go further in identifying the most stable of these IPs over the collection period, such that those that are purely illicit are targeted for ISP takedowns, while the benign ones (e.g., CDNs) are advised to improve their security practices.

| Maximum Signature Length | Day-Pair 1 | Day-Pair 2 |
|---|---|---|
| 1 | 32.5% | 38.1% |
| $\lfloor log_2(X) \rfloor$ | 41.1% | 46.7% |
| 2 | 46.5% | 51.7% |
| 3 | 48.0% | 53.6% |
| 4 | 48.2% | 53.7% |
| **5** | **48.2**% | **53.8**% |
| 10 | 48.3% | 53.8% |
| 20 | 48.3% | 53.9% |
| 50 | 48.3% | 53.9% |
| 100 | 48.3% | 53.9% |

**Table 4.4:** Sensitivity analysis. $\lfloor log_2(X) \rfloor$ is the variable length signature with the size being the rounded-down logarithm of the component size $X$.

### 4.3.2 Longitudinal Analysis

So far, I have looked at the malicious file delivery ecosystem over 24 hours. However, many questions remain unanswered on how such delivery ecosystems evolve. Therefore, in this section, I analyse the temporal evolution of file delivery networks, particularly focusing on the retention rates of infrastructure.

#### 4.3.2.1 PUP Ecosystem persistence

First, a graph is built for each day. As a result, the PUP Ecosystem (i.e., Giant Component) was found to be stable over the entire year. This result is important, as prior work [127, 131] only characterises PUP and malware ecosystems in the short-term. As described in Section 4.2.2.1, the network-only and file-only components from the overall graph are then computed, which represent the network-based and file-based delivery infrastructures.

#### 4.3.2.2 Infrastructure tracking

It is important to develop robust signatures to track infrastructures in time. As such, a graph percolation experiment was conducted to measure how quickly the GC breaks down using a number of graph influence measures, i.e., eigenvector, betweenness, in-degree, out-degree, and overall degree centralities (see Figure 4.3). Following this experiment, I select out-degree as the criteria to select influential nodes for infrastructure signatures (see Section 4.2.2.2). In practice, degree and out-

degree perform identically, but out-degree is selected as it is more computationally efficient in that it does not include (redundant) leaf nodes in the tracking signature.

I conduct a sensitivity analysis of tracking performances with different maximum signature lengths. Here, I select infrastructures from two randomly selected pairs of consecutive days in the data series (i.e., Day-Pair 1: 2015-Oct-22 and 2015-Oct-29, and Day-Pair 2: 2016-Feb-02 and 2016-Feb-09). A match is defined as an intersection of a pair of signatures across two days. I then compute the percentage of component signatures matched across these day-pairs using different signature lengths. Finally, by diminishing returns, I select a maximum signature length of 5 (see Table 4.4).

This result means that a graph component can be characterised by *up to* five of its top out-degree nodes. An example of a network component signature is {*'http://groupsetzipmyjob(dot)org/hp/'*, *'107.21.97.98'*, *'54.225.102.164'*, *'68.232.34.200'*, *'74.120.16.179'*}. Besides making tracking computationally feasible, this also points out elements (e.g., IP addresses, DNS domains) that are stable over time and could, therefore, constitute potential intervention points by law enforcement agencies (LEAs) and security companies (e.g., for takedowns).

In the tracking analyses, I only consider file clusters that exhibit dropping behaviour as file (client-side) distribution infrastructures. The retention rates of the remaining *lone* files is considered separately as these are less easily attributable to individual actors. I also track infrastructures using two temporal granularities: daily (over a month) and weekly (same weekday sampled over a year). This approach allows us to observe in detail the delivery network life-cycles in both the medium-term and the long-term trends.

#### 4.3.2.3 Retention of infrastructures

Figure 4.10 shows the daily retention of network and file delivery infrastructures, i.e., the number of infrastructures that are detected from one day to the next. The daily retention reveals cyclicity in the network and file distribution infrastructures, both that are active in download activity (total) and that are tracked, with a cyclic period of seven days. As 1st October 2015 was a Thursday, the results show that more

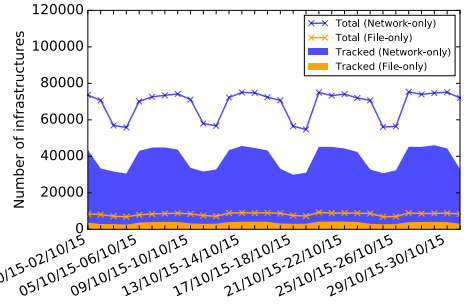


**Figure 4.10:** Daily retention of delivery infrastructures over a month.

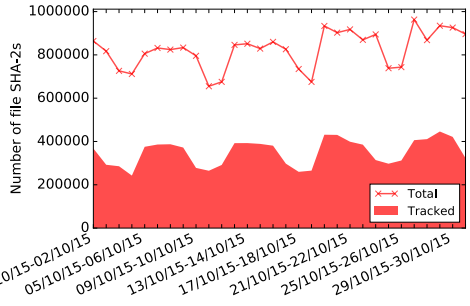


**Figure 4.11:** Daily retention of lone file SHA-2s over a month.

distribution infrastructures are active across weekdays (i.e., Mon-Tue, Tue-Wed, Wed-Thu, and Thu-Fri) and less across weekends (i.e., Fri-Sat, Sat-Sun, Sun-Mon). The cyclic download activities during the week could show that the file distribution patterns of cybercriminals and legitimate providers alike mirror the network use, download, and work-rest patterns of people and organisations. In other words, infections increase during business hours because more potential victims have their computers on, as already observed by previous work [70, 186, 188]. *Routine Activities Theory* [68] supports this notion, which posits that (cyber)criminals can only engage in criminal activities when they converge in (cyber)space and time with suitable targets in the absence of capable guardians (or cyber defences).

Figure 4.11 shows the daily retention for lone files. A lone file is a file that is dropped directly from network hosts and does not engage in any further dropping behaviour. On the other hand, I defined a file (client-side) distribution infrastructure as a file-only component that exhibits dropping behaviour between files. In comparison with the retention of file infrastructures (Figure 4.10), the fluctuations in the presence of lone files (Figure 4.11) and network infrastructures (Figure 4.10) appear more pronounced. This is probably due to there being many more network infrastructures and lone files than file infrastructures, e.g., lone files constitute 98% of file-components in the first graph snapshot. It should be noted that the weekly fluctuation in the total and tracked network infrastructures may not specifically represent hosts going down or coming up: it only means that the sensors used in this dataset do not observe downloads from these hosts.

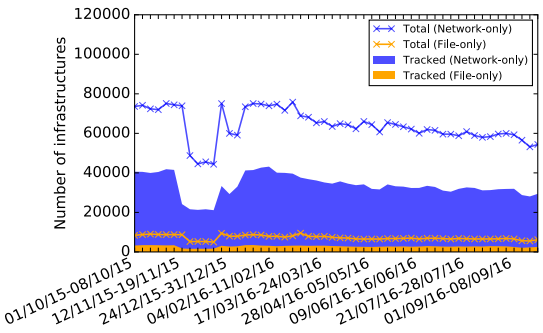


**Figure 4.12:** Weekly retention of delivery infrastructures over a year.

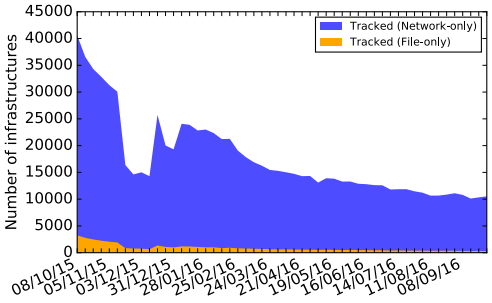


**Figure 4.13:** Lifespan of delivery infrastructures tracked from 1st October 2015.

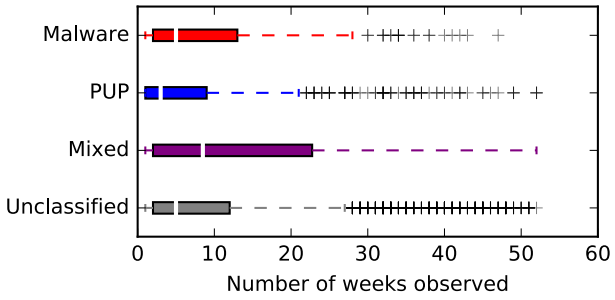


**Figure 4.14:** Box plots showing the lifespan of file delivery infrastructures.

As shown in Figure 4.12, the weekly retention of delivery infrastructures (sampled every Thursday for a year), omits this weekly periodicity. However, there is a large drop in download activity from 19th November 2015 until 14th January 2016, with a small peak in activity on the week of 17th to 24th December 2015. I later investigate this anomaly (see Section 4.3.2.5).

#### 4.3.2.4 Lifespan of infrastructures

Figure 4.13 is the lifespan plot of distribution infrastructures observed since the first day of analysis (1st October 2015), with a weekly granularity. This figure shows the activity decay of these infrastructures over a year. That is, infrastructures observed on each sampled day are matched with infrastructures observed on 1st October 2015, where the sampling frame is seven days. I initially track 40.6k network infrastructures and 3.2k file infrastructures and find that, of these, at least 30k network infrastructures (75%) remain active for over 6 weeks, while 10.5k network infrastructures (26%) and 320 file infrastructures (10%) remain active for a year.

There is also a several-week dip in activity starting some time between 12th and 19th November 2015. However, the rise in tracked infrastructures between 17th and 24th December 2015 indicates the re-emergence of some of the network and file infrastructure activity that was lost. This volatility in network and file delivery infrastructure activity could be the result of these infrastructures going in and out of service (e.g., server take-downs, technical faults, cessation of activities, new actors entering the ecosystem). However, this could also be hosts utilizing fast flux or DGA and/or prolific droppers undergoing polymorphism. An additional measurement was conducted on the stability of delivery infrastructure in Appendix A.1, indicating an even smaller proportion of nodes being stable *during* the whole year.

Figure 4.14 shows the lifespan of file delivery infrastructures that drop only PUP, only malware, or both. I identify and track 344 confirmed malware-only delivery infrastructures, 805 PUP-only ones, and 50 infrastructures that deliver both PUP and malware. As shown in the figure, client-side (not network-level) delivery operations involving malware appear to be longer-lasting than PUP ones, i.e., mawlware file-dropping networks are active for a median of 5 weeks vs. 3 weeks for PUP. This could be due to the possibility that malware is stealthier in their installation and operation on a victim computer, and/or more resilient to removal than PUP. Mixed operations involving both PUP and malware appear to be the most enduring. However, the validity of this result is questionable as this class has a small sample size.

#### 4.3.2.5 Case study: Dyre and the anomalous drop in activity

A large drop in download activity occurs between 12th and 19th November 2015. Symantec [196] report the virtual cessation of activities of the cybercriminal group that controlled the Dyre financial fraud trojan, following a Russian law enforcement operation in November 2015. In particular, they report an abrupt halt in Dyre-related email spam campaigns from 18th November. Symantec also confirm the drop of its associated malware, such as the `upatre` dropper family, which is the dropper malware that is installed onto victim computers (often by victims downloading malicious email attachments) before downloading the Dyre trojan.

Using the AVClass labels of files observed on 12th and 19th November, I correlate the changes in the presence of malware and PUP families with this event. I do not observe any `dyre` family labels within this period, but interestingly, I observe significant drops in the presence of other families. I find a significant drop in `upatre` – 44 out of its 99 SHA-2s cease in activity. However, most interestingly, the largest disappearances are in the popular PPI droppers: `amonetize` ($-595$ SHA-2s), `installcore` ($-374$ SHA-2s), `eorezo` ($-266$ SHA-2s), and `convertad` ($-214$ SHA-2s). I also find a drop in the `neshta` malware ($-223$ SHA-2s) These changes are significant, seeing as there was only a total of 8k file SHA-2s with known family labels the week just before the drop in activity.

I repeat the same difference analysis of observed families across the period of 17th and 24th December. I discover a recovery of some sorts of `amonetize` ($+445$ SHA-2s), as well as an order of magnitude smaller increase in some of the other popular PUP families: `opencandy` ($+58$ SHA-2s), and `eorezo` ($+47$ SHA-2s). However, 5 `upatre` dropper SHA-2s disappear during this period, with only a remainder of 36 SHA-2s being observed.

Though there is evidence of a significant drop in the presence of the Upatre dropper over this period (which could have been as a result of the reported law enforcement operation against the Dyre operation), the coinciding drop in activity of prevalent PUP and popular PPIs, such as Amonetize and others, during this same period is interesting. Perhaps these PPI services were patrons to the Dyre gang. Alternatively, perhaps there was some shared network infrastructure (such as ISPs) which hosted the services that distributed these two types of unwanted software.

**Global drop in download activity.** Investigating this phenomenon a little further, it was found that (i) the drop in download activity was felt across all categories of software (Figures 4.15 and 4.16), and (ii) the download activities of certain groups of domains were affected more than others (Figure 4.17) – for example, `.ru` domains suffered a less significant drop in activity than `.com` domains. Why this occurs is not known. However, contrary to some of the insights drawn from the Dyre case study above, this finding presents confounding evidence against any special links

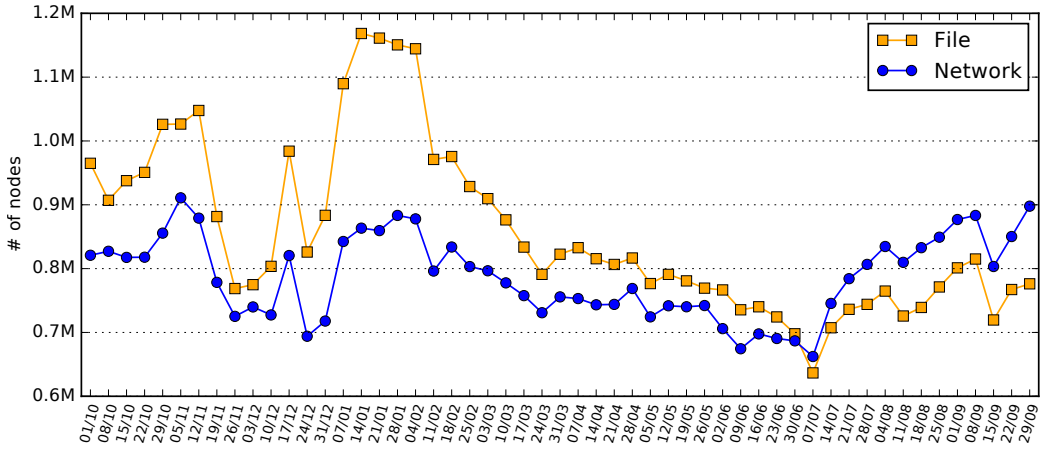


**Figure 4.15:** Overall numbers of file nodes and network nodes observed over a year.

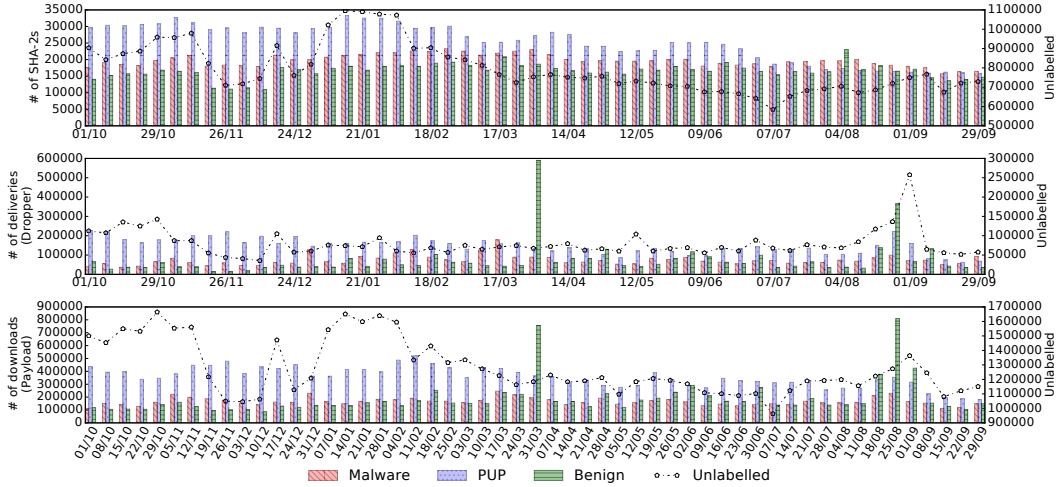


**Figure 4.16:** Download metrics of different classes of software over a year.

between the Dyre-Upatre operation and other unwanted software (Amonetize, Installcore, etc), seeing as many other software families experienced similar drops in activity. This indicates the need for further research to understand this anomalous global drop in activity and the complex dynamics between the software families involved.

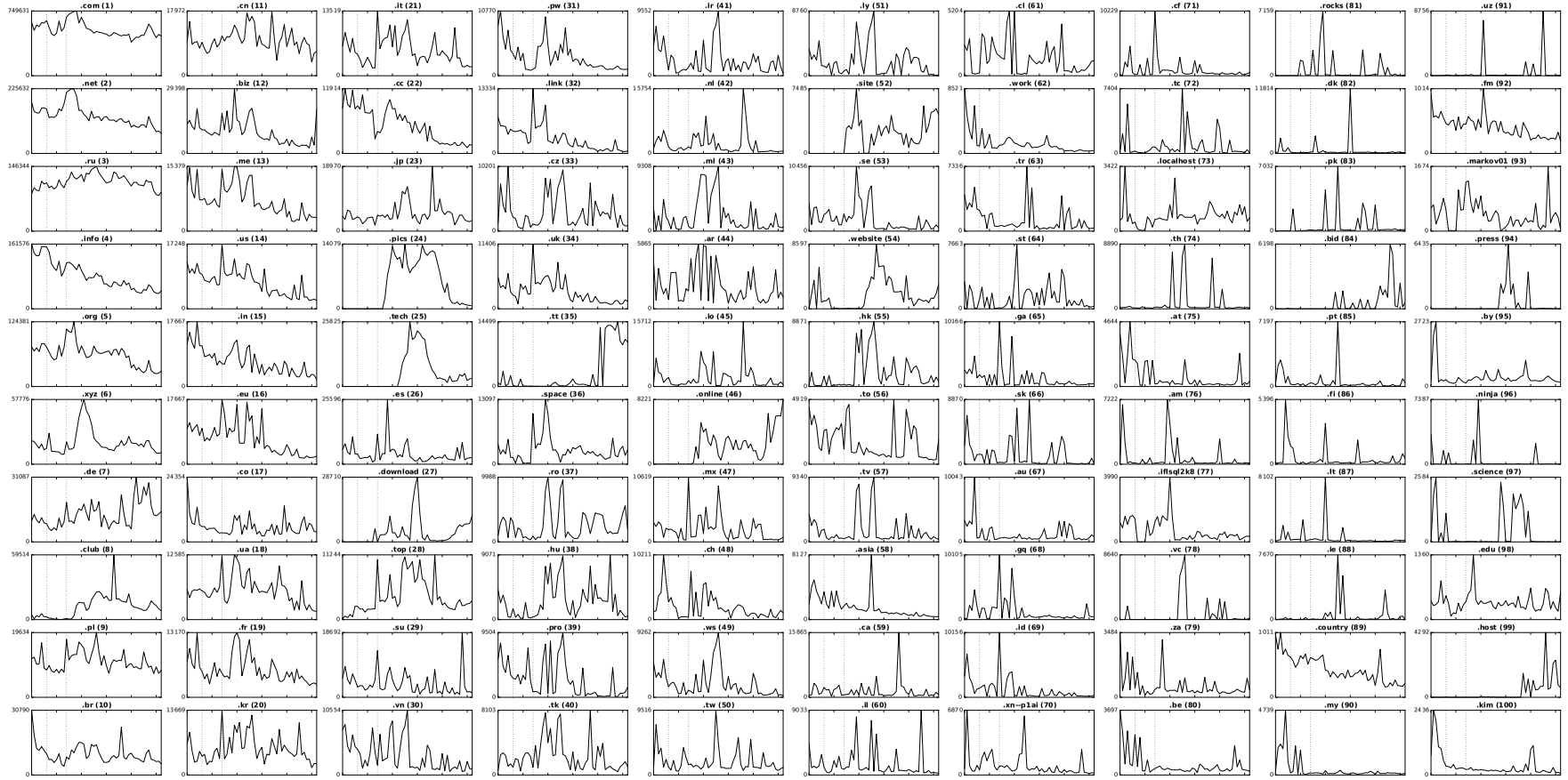


**Figure 4.17:** Download activity of top 100 TLDs from October 1st, 2015 to September 29th, 2016.

#### 4.3.2.6 Summary of results

In this longitudinal analysis, I showed that the PUP Ecosystem is stable in the long-term. I showed periodic download patterns over a week, perhaps in accordance with the Routine Activities Theory from criminology [68]. I also show that network infrastructures tend to be quite short-lived, where 75% survive for over 6 weeks, while 26% survive for over a year. Finally, a case study was presented on a global drop of download activity across a wide array of software brands and families, but coincided with a takedown operation against the Dyre malware. Though it is posited that there could be common distribution backbones between malware, such as Dyre, and popular PUP PPIs, such as Amonetize, the finding that many other families also experience a coincident drop in activity during this period throws this hypothesis into question. This indicates the need for further research into this phenomenon.

## 4.4 Discussion

In this study, I presented a data-driven analysis of the delivery of malicious files on the Web. These findings shed some light on malware and PUP operations more comprehensively than previous work. In this section, I take a step back and reason over what these findings mean, and how they could be applied for mitigation purposes. I then highlight some limitations to this study.

### 4.4.1 Implications of Findings

I found two largely disjoint ecosystems: one responsible for the delivery of PUP, and one dedicated to installing malware on victim computers. I discover that the malicious file delivery ecosystem makes considerable use of CDNs, which can make takedown operations difficult. On the other hand, I identified ASes in which malicious network infrastructures congregate. This result is consistent with previous research [190] and suggests that ISP-based interventions can still be a valid method to disrupt malware operations. In this study, I considered a methodology to identify network elements (DNS domains, IP addresses) that do not change over time. This methodology could be further developed to identify optimal intervention points that LEAs could target to perform disruption, solving the fundamental problem of iden-

tifying the right elements to target when performing takedowns, as highlighted by previous work [150]. Other future works include repeatability experiments with other (open-source) datasets, and identifying more real-world stimuli (e.g., ISP takedowns) and MDN adaptations within the data.

### 4.4.2 Limitations

As mentioned, this data-driven analysis has limitations. By applying graph analysis techniques to the download graph, I obtain a proxy to what is happening on the victim computers. The type of analysis that I perform allows me to characterise the operation of PUP and malware delivery networks, but one cannot be certain about some of the details of malware operations that go beyond the study data. For example, by looking at the dropping behaviour of hosts, one may estimate whether they belong to exploit kit infrastructures or not. However, one cannot observe the actual vulnerabilities being exploited on the host as part of a drive-by download attack. For this reason, it could be that some of the infrastructures that one considers exploit kits are just relying on social engineering. In a similar sense, one cannot see auxiliary connections between hosts, such as direct web links.

I identify files using their SHA-2 (256 bits) hash function. This allows me to reliably distinguish between unique files, e.g., variants of the same malware family, or to identify the same file in the wild under different guises, e.g., a malware binary using different file names. However, this method of identification still presents complications for packed files. Packing alters the SHA-2 of a file and so the same binary that is re-packed multiple times would appear as different unique files. This may manifest in the graph as a host or dropper delivering multiple files when they are actually repacked versions of the same file binary. This raises the need for some additional file clustering techniques.

As an additional limitation, some of the analysis relies on third-party information such as VirusTotal and the AVClass tool. This information, however, is not perfect (e.g., some false positive indications), and, as I have shown, is often incomplete. For this reason, some of the file components that I identified may have been misclassified. I focus my analysis on files with known families. This helps to miti-

gate false positive indications from VirusTotal, as each binary in this dataset that is assigned a family name by AVClass has at least 2 different AV engines agreeing on the associated malware/PUP family. This classification excludes AV engines that may also assign positive indications, but are not taken into account by AVClass due to them only assigning a generic family name.

## 4.5 Conclusion

In this study, I presented the first comprehensive data-driven analysis of malicious file distribution on the Web. I showed that there are two disjoint ecosystems responsible for the delivery of PUP and malware, respectively, and that the PUP ecosystem is particularly stable over the long-term. Studying the characteristics of these ecosystems in detail, together with their temporal dynamics, I showed that the PUP ecosystem is responsible for 80% of suspicious downloads worldwide. I estimated the ratios of PUP-to-malware in the wild to be 17:2 and differentiated the modus operandi of file distribution between the two ecosystems. I also tracked these distribution infrastructures over a year, finding that 75% of malicious network infrastructures survive for over six weeks, while 26% survive for over a year. These findings help researchers gain a better understanding of this ecosystem, and allow us to identify promising routes for more effective mitigation against the distribution of malicious software. For instance, a methodology was devised to identify those elements in a delivery infrastructure that change slowly over time. In future work, one could explore the possibility of using such elements (IP addresses, autonomous systems, domain Names) for performing effective takedown operations.

# Chapter 5

# Tracing the Evolution of Malware Delivery Operations Targeted for Takedown

In this chapter, I present my second experimental study, which is an extension of the measurement study presented in the previous chapter. However, the focus of this study is to analyse the behavioural dynamics of three malware delivery operations after law enforcement attempted takedowns against them. This work was in collaboration with *Symantec Research Labs*. We published a RAID conference paper based on this chapter titled, *'Marked for Disruption: Tracing the Evolution of Malware Delivery Operations Targeted for Takedown.'* The analysis code is publicly available on GitHub.[1]

## 5.1 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

[1] `https://github.com/ColinIfe/mdn`

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 [192, 188, 2, 33, 22]. Even worse, these botnets could be further monetised as *pay-per-install* services [47], allowing the botnet controller to rent out access of their network to other criminals and their malware.

As described in Section 1.1, cybercriminals have devised numerous evasive techniques to avoid detection and make their malware operations more resilient. For example, on the software level, polymorphism is one technique that is used to beat antivirus detection engines, where malware constantly changes its identifiable features to make each binary appear different from one another [31]. On the network level and to avoid detection of their malicious servers, cybercriminals may employ Fast Flux – the rapid changing of the public IP address of a given server [109]. Alternatively, they may use domain generation algorithms (DGAs) – hard-coded algorithms in the malware that enable them to alternate between the C&C domains with which they communicate [27]. Furthermore, cybercriminals have been known to use distributed server architectures, having servers hosted in multiple geographic regions and across different autonomous systems. This could be a tactic to avoid detection (i.e., delivering malware from different sources) [172] or to ensure there is redundant infrastructure available in the event of a takedown [137].

As I outlined in Section 1.2, law enforcement agencies (LEAs), security companies, and researchers are constantly seeking methods, opportunities, and intervention points to disrupt the serious and growing threat of botnet and malware delivery operations [150, 83]. Takedown operations are just a subset of some of the disruptive techniques that may be 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 they are hosting; or physically seizing malicious server infrastructure and assets, and arresting the miscreants

involved. These various takedown strategies were described in more detail in Section 2.6. The success of such counter-operations is mixed [83].

Although the details of a number of takedown operations have been recorded in the literature, very little examines how the targeted malware delivery operations actually respond after such interventions. This leaves many important questions unanswered. For instance, 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 these takedown operations, are there other intervention points in the malicious infrastructures that could prove to be more effective targets? Finally, considering the behaviours of these miscreants, could some of these reactions be predicted and taken into account by LEAs and security practitioners?

In this study, using global download metadata collected in October 2015–September 2016, I devise a novel tracking and analysis methodology to quantitatively assess the evolution of malware delivery operations that are targeted for LEA takedown. In particular, I focus on three malware delivery operations (botnets) that were targeted for takedown 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. I then track and comprehensively analyse the activities of these malicious operations over the course of a year from multiple perspectives. More specifically, I conduct this analysis by breaking down the malware delivery operations into two components: the upstream network infrastructure (server-side) and the downloaded binaries and their dropper networks (client-side). This analysis paints a detailed picture of the dynamics, complexities, and business relationships of malware delivery operations, particularly in light of a takedown attempt, and provides the security community important pointers to consider when effecting future takedowns. In summary, this study makes the following contributions:

- I provide 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.

- I observe a myriad of behavioural responses to takedown attempts by each malware delivery operation. Specifically, I show 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* [105] and *defiance* [180] 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 of malware delivery operations, while also uncovering challenges and further opportunities to disrupting them.

## 5.2 Targeted Malware Delivery Operations

In this study, I seek to uncover how specific operations evolve as a result of law enforcement takedown operations against them. In order to carry out such an analysis, it is important to first identify takedown operations that occurred within the dataset collection period, i.e., between 1st October, 2015 and 29th September, 2016. Second, one would need to ascertain the ground truth available for any given botnet operation, and whether it would be enough to provide any further insight to how these botnets respond. As such, in this section, I outline three different botnets identified as targets for takedowns within the period of the dataset, on which this study focuses: 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. It has also been known to self-replicate by copying itself from compromised devices to mapped network drives and local storage devices [4], as well as be delivered through exploit kits on compromised web servers [9]. In late 2015, an indictment issued by the United States FBI stated that the Dridex operation was reported to have caused losses of over $10 million in the United States alone, and over $25 million worldwide [3, 14].

**Ground truth.** Of all the malware delivery operations studied in this work, the Dridex case study appears to have the most ground truth available in the public domain. Namely, in addition to information gleaned from news reports and technical assessments of the malware, the United States' FBI and Department of Justice released a total of ten unsealed court documents relating to the Dridex investigation and consequent sinkhole operation[2]. This ground truth is summarised as follows:

- In August 2015, one botnet administrator was arrested in Cyprus, while four other co-conspirators, believed to be amongst the botnet leadership and located in Russia, remained at large [4, 3, 14].

- On 4th September, 2015, the National Crime Agency in the United Kingdom undertook a takedown operation to collectively disable the C&C servers that formed the backbone of the Dridex botnets [3, 14]. As a result, the super-peers and peers of these botnets were believed to no longer have any centralised mechanism from which to take direction and receive new commands. Nonetheless, it was understood that the remaining operators could still re-establish contact with these bots and continue their nefarious operations.

[2]The full legal documents are accessible online at `https://www.justice.gov/opa/pr/bugat-botnet-administrator-arrested-and-malware-disabled` and `https://www.fbi.gov/contact-us/field-offices/pittsburgh/news/press-releases/bugat-botnet-administrator-arrested-and-malware-disabled`

- On 9th October, 2015, law enforcement officials began a **60-day *DNS sinkhole* and *disinfection*** intervention [3, 14]. In this study, I estimate that the counter-operation occurred in the window between the ***8th October and 10th December*** observations. The specifics of the DNS sinkhole operation were sealed, but it was known to target the super peers of the botnet, such that the C&C servers would not be able to communicate with already infected end-user computers. It is also unclear which geographic regions were affected by this DNS sinkhole, but it is likely that the law enforcement agencies placed greater weight on servers based in the US. The disinfection operation involved the authorities contacting victims providing instructions on how to remove the Dridex malware from their devices.

- In mid-December, 2015, at the end of the 60-day intervention, a court order was renewed to extend the disinfection for an unknown duration [3, 14]. It should be noted that I only depict the 60-day intervention in this analysis since it is a known and strictly-defined period.

## 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 [13, 17]. They have been known to propagate through infected USB flash drives, instant message applications, social networks, spam messages, and exploit kits. It has also been noted that, at some point, a significant proportion of the Dorkbot infrastructure was based in Poland [17]. However, at the time of the takedown operation, it had diversified to having C&C servers in other regions, such as the rest of Europe, Asia, and North America [12].

**Ground truth.** Like the takedown operation against the Dyre and Upatre botnets, there is limited open source intelligence regarding the Dorkbot takedown. The following ground-truth is available from public sources:

- In December 2015, security companies and law enforcement bodies around the world conducted a swift ***DNS sinkhole*** **and** ***seizure*** **operation** against the Dorkbot botnet [15, 18, 12].

- The precise date of the takedown operation is unknown, but it is estimated to have occurred on or just before 3rd December, 2015, which was the when the takedown operation was first announced in the public domain retrospectively [18]. Further, Microsoft confirmed that this takedown occurred early in December [15]. Therefore, in this study, I estimate that the counter-operation occurred in the window between the ***26th November and 3rd December*** observations.

## Dyre and Upatre

The Dyre and Upatre operations provide an interesting case study, not least given the reported law enforcement operation targeting the Dyre botnet coincided with a sudden, global drop in malicious download activity, which was observed in an earlier study [116]. Dyre (also known as Dyreza, Dyzap, and Dyranges) is a sophisticated financial fraud trojan that targets Windows computers. Dyre is designed to steal credentials and hijack banking sessions on infected machines in man-in-the-middle fashion, siphoning off funds from the compromised accounts to those controlled by the botnet operators. Dyre has also been reported to use infected machines to replicate itself and send copies to further users through the victim's email contact list [11]. 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, Upatre proceeds to install Dyre malware onto these devices [11, 5, 189]. 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 [19].

In this study, I focus only on the activities of the Upatre dropper. This is because little to no observable Dyre download activity is found in this dataset, hence giving little insight on its evolution. Why exactly this is the case is not known. However, since Dyre was known to undergo rapid polymorphism [11], 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.

**Ground truth.** Open source intelligence on the law enforcement operation against the Dyre-Upatre operations is very limited, especially with regards to the particulars of this counter-operation. Nonetheless, sources have established the following:

- In November 2015, law enforcement officials conducted a ***seizure* and *arrest* operation** against the Dyre operators in Moscow, Russia.
- The precise date of this counter-operation is unknown, but it is estimated to have occurred on or just before 18th November, 2015, which is the day sudden drops in Dyre and Upatre activity were observed by security researchers [5, 116]. Furthermore, sources reported that the arrests occurred between 18th and 19th November [42]. Therefore, in this study, I estimate that the counter-operation occurred in the window between the ***12th and 19th November*** observations.

## 5.3 Methodology

Although the principal focus of this study is on the dynamics of three specific malware delivery operations in light of law enforcement takedowns against them, I devise a generic methodology that may be adopted to characterise any class of file delivery operation, whether it be malicious or benign. Therefore, in this section, I detail the steps taken to (i) build the download graphs for the year-long dataset; (ii) classify the file nodes as either malware, potentially unwanted program (PUP), 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, I only seek to analyse file delivery *operations* – not file delivery *campaigns*. More precisely, I 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 [183]). I 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.

### 5.3.1 Building Download Graphs

I adopt the graph-building methodology as described earlier in Section 4.2.1.2. For ease of reference, the structure of a download event is outlined in Equation 5.1:

$$\mathbf{d} = < I, D, U_r, ..., U_f, F_f, A_f, U_p, F_p > \tag{5.1}$$

where $I$ is the IP address from which the file was downloaded, $D$ is its FQDN, $U_r$ is the initial URL in an HTTP redirection chain, $U_f$ is the host URL of the download (after removing the URL parameters) and the terminal URL in a redirection chain, $F_f$ is the downloaded file identified by its SHA-2, $F_p$ is the parent file identified by its SHA-2, and $U_p$ indicates the URL from which this parent file was downloaded. $A_f$ represents a set of attributes which provides additional information about file $F_f$, such as its filename, its size (in bytes), and the "reputation" and "prevalence" scores assigned to these files by Symantec's static and dynamic analysis systems (see Section 3.2.1).

It should be noted that an updated graph schema is used in this study as opposed to the one used in the measurement study of Chapter 4. In particular, this new schema includes nodes that denote fully qualified domain names (FQDNs). This schema also retains download event information for every node in the graph. An example of the new schema is shown in Figure 5.1.

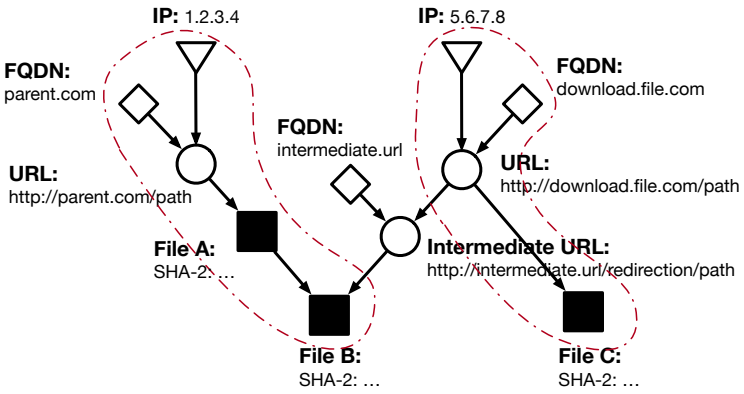


**Figure 5.1:** An updated schema to interpret download graphs, which now includes FQDNs. Two series of download events are highlighted.

### 5.3.2 File Classification

Having constructed the download graphs for each observation window, I build on the file classification technique used in Section 4.2.1.4. Specifically, each file node (based on its SHA-2) is labelled as either *malware*, *potentially unwanted program (PUP)*, *benign*, or, if there is no available ground truth, *unlabelled*. If it is known, I also specify the *software family* to which the SHA-2 belongs, whether malicious or benign. Otherwise, I label SHA-2s without known software family labels as *singletons*. In total, I 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). I later track the evolution of delivery operations associated with each malware or PUP family in time, while also retaining benign and unlabelled files to observe the co-evolution of malicious activity in the context of background downloads.

#### 5.3.2.1 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, I configure the AVClass

tool to map specific aliases to specific families. Specifically, based on the sources for each malware operation in Section 5.2, I aggregate the following aliases to each respective family:

- 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.

### 5.3.3 Tracking and Analysing Operational Activity

Besides just monitoring malicious file presence, I 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 utilised against them, such as botnet takedowns or coordinated arrests. This goal is achieved in two stages. First, I devise a methodology to identify and track a (malicious) file delivery operation. And second, I derive a set of metrics that describe different aspects of a given file delivery operation, and conduct time series analysis on these metrics.

#### 5.3.3.1 Tracking Delivery Operations

The method to tracking delivery operations and their activity is simple. That is, for a (target) software family that I seek to analyse, $SF$, and for the $i$th observation period, where $i \in [1..53]$ (i.e., every Thursday for a year), the following algorithm is used:

1. Compute $F_i^{SF}$: the set of all file nodes pertaining to software family $SF$ in observation period $i$.
2. 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 $i$. These parent nodes represent part of the upstream delivery network

used to distribute software family $SF$ directly. 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 [47].

3. Compute $C_i^{SF}$: the set of all child nodes (files) that are dropped by the files in $F_i^{SF}$ in observation period $i$. These child nodes represent part of the downstream delivery network of software family $SF$, provided that this software family downloads other files. Being payloads, these child nodes could be attributed to the clients of the $SF$ delivery network.

4. Finally, compute the node attribute look-up table, $A_i^{SF}$, which stores the attributes of all file nodes (target, parent, child) and network nodes (URL, IP, FQDN) that form the delivery network of software family $SF$ in each observation period $i$. File node attributes include software family, file size, reputation and prevalence scores, # of times downloaded, and # of drops. Network node attributes include location (country), top-level domain, and effective second-level domain, as applicable.

#### 5.3.3.2 Time Series Analysis

I seek to generate metrics (or features) which sufficiently describe the different aspects of a file delivery operation over the observation period. Using the amalgamated data structures, $F^{SF}$, $P^{SF}$, $C^{SF}$, and $A^{SF}$ as defined above, I compute and analyse time series data based on two groups of metrics:

**Network dynamics**. This group of metrics capture the dynamics of the server-level activity in the file delivery operation. The numbers of URLs, domains, IPs, and countries used to host the delivery servers and deliver files to end-users, indicating the pervasiveness and extent of resources used for the delivery operation. The numbers of IPs associated with each domain provide indicators of the possible use of the Fast Flux technique (rapidly changing IPs) [109], or the use of content distribution networks (CDNs) and servers spread across different geographic regions – common methods to avoid detection and to increase botnet resilience [172]. On the

| Group | Metric | Description |
|---|---|---|
| Network Dynamics | | *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. |
| Downloader Dynamics | | *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 5.1:** The metrics used to analyse each malware delivery operation.

other hand, the number of domains associated to any given IP could be indicators of servers residing within shared-hosting clusters, or servers using domain generating algorithms (DGA) – another commonly used technique by botnet C&C servers to avoid detection [27, 163]. Finally, I also quantify the most popular domains, top-level domains (TLDs), IPs, and countries used for each delivery 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, I obtain the total and per-SHA-2 counts of download and dropping events for the software family, which are key performance indicators of its delivery operation. I also keep track of the total and top $N$ families involved in the software family's download activities. Further, I analyse the numbers of URLs, domains, and IPs used to deliver each file SHA-2, which are all indicative of the diversity in distribution vectors used, perhaps to increase outreach to end-users, or to evade

detection systems more effectively. I also examine metrics that indicate polymorphism (a malware characteristic that is used to evade detection [31]): 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 security systems). It should be noted that a higher malice score corresponds with a higher likelihood that a file is malicious, while a higher prevalence score indicates that a file is observed by Symantec sensors more frequently (see Section 3.2.1).
All these metrics are summarised in Table 5.1 and analysed in Section 5.4.

**In summary,** this operation tracking and analysis methodology, coupled with the labelled, longitudinal graph data, grants 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.

## 5.4 Analysis

In this section, I apply the techniques as described in Section 5.3.3 to analyse the evolution of three different malware delivery operations: the Dridex, Dorkbot, and Dyre-Upatre botnets. Each of these botnets faced law enforcement agency (LEA) takedown attempts between October 2015 and September 2016. For each malware delivery operation, the time period of the associated LEA operation is highlighted, allowing one to analyse the evolution of the botnet's activities in light of this counter-operation. The analysis of each malware delivery operation is broken down into two general categories of metrics: its network dynamics, and its downloader dynamics.

### 5.4.1 Network Dynamics

I begin my analysis with the upstream delivery networks of each malware delivery operation, where I compute the network dynamic metrics as described in Section 5.3.3.2 and analyse them herein. Figure 5.2 shows a number of time series denoting aggregate network dynamics, and Figure 5.3 evasion indicators for each malware delivery operation. I note some apparent features. For instance, Dridex exhibits consistent growth in all forms of network activity from early October 2015

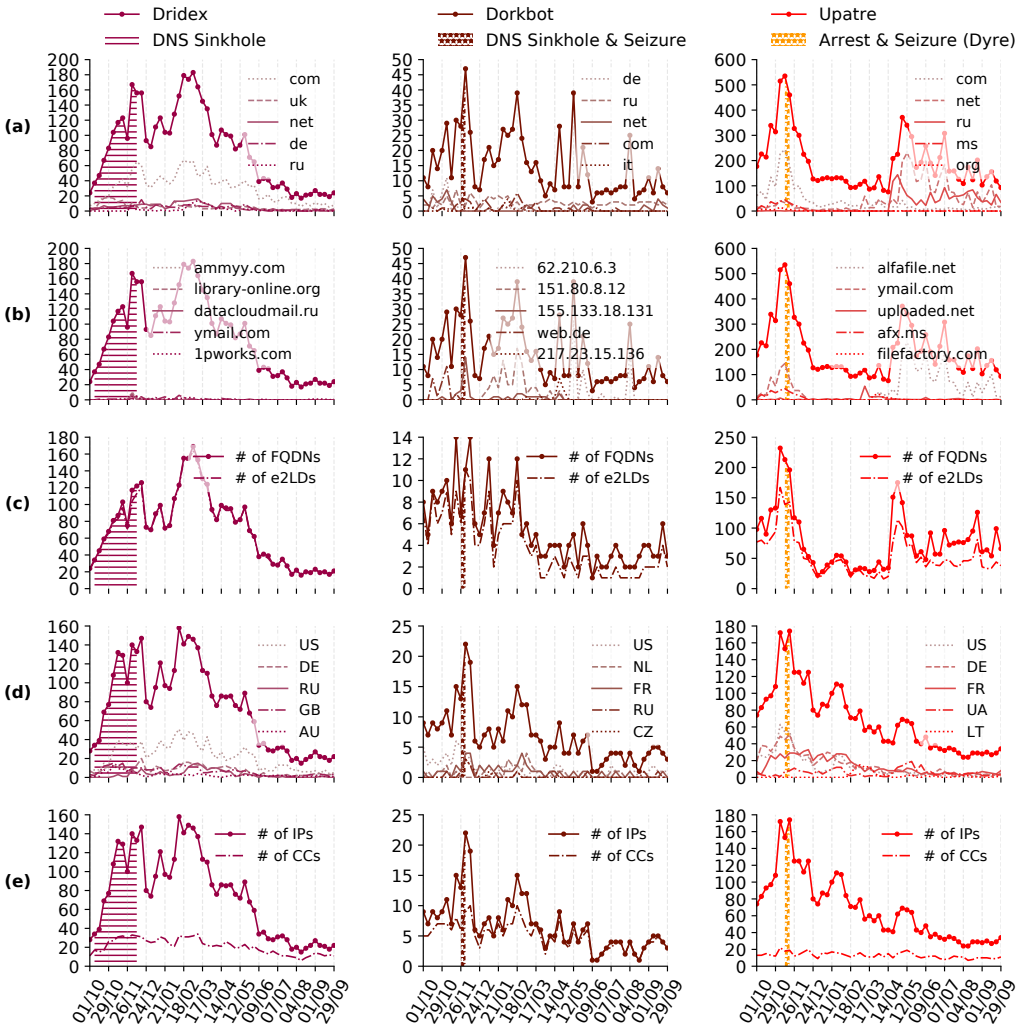


**Figure 5.2: Aggregate network activity:** **(a)** # of URLs used and top 5 TLDs; **(b)** # of URLs used and top 5 e2LDs/IPs; **(c)** # of FQDNs and # of e2LDs; **(d)** # of IPs used and top 5 hosting countries; and **(e)** # of IPs and # of hosting countries. Dridex exhibits consistent growth in network 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.

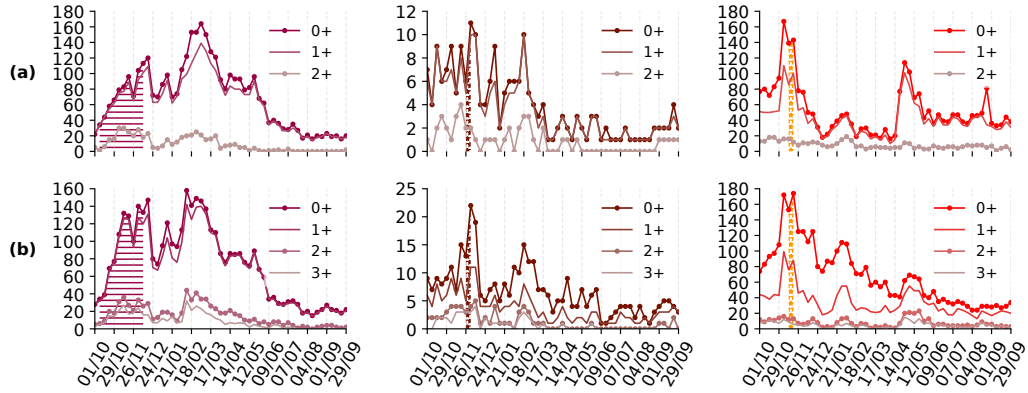


**Figure 5.3: Evasion indicators: (a)** # of e2LDs associated with $N+$ IPs; and **(b)** # of IPs associated with $N+$ e2LDs. Dridex was found to use shared-hosting platforms and CDNs often. Upatre increases its use of IPs with 2+ domains from mid-April, most of which were for `.ru` DGA domains.

(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 [83] 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.

#### 5.4.1.1 Dridex

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

The first observation is that the period of consistent growth in malicious server activity seems to be aligned with the same 60-day period of the DNS sinkhole counter-operation. 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 [83]. This is clearly not the case for the Dridex operation, where we observe the opposite: an increase in Dridex network activity both during and following the takedown operation. 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 and subsequently missed 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 LEA 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 [3]. As a result, and perhaps in retaliation, the botnet operators may have increased their activities and/or moved their operations elsewhere, both of which could lead to an overall increase in network activity during this period.

The second observation of interest is that when we look at the actual number of download/redirection URLs used in the first era (in conjunction with the most common TLD suffixes) and the number of IPs used, as shown in Figures 5.2(a) and 5.2(d), respectively, we see significantly increased usage of download URLs with a `.com` suffix and download servers hosted in the US. 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 primarily 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, so they would just be attempting to recover lost ground. Nonetheless, it is clear that these malware operators ramped up their operations at the same time that LEAs 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 indicated by the low proportion of download activity by the most commonly used domains (`ammyy.com`, `library-online.org`, etc), as shown in Figure 5.2(b). This is also reflected in the approximate 1:1 ratio in # of FQDNs-to-# of e2LDs attributed to its download servers (Figure 5.2(c)). Similarly, as Figure 5.2(e) shows, up to 35 different countries are used to host Dridex download servers. Querying the data, I found that the Dridex operation makes significant use of (i) websites on common, 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, respectively, as shown in Figures 5.3(a)–(b). This diversification in distribution channels naturally makes it difficult to identify bottlenecks in the Dridex operation. It is possible that this approach was 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. It also appears that this reduced server usage stabilises for a few weeks from around 4th August. Without additional data, it is difficult to draw any robust conclusion on the potential causes of this reduction in network activity, particularly on the likelihood that it was as a consequence of the takedown operation, a second (undocumented) operation, or some other factor. However, I also note how its net-

work activity stabilises from 4th August. One cannot rule out the possibility that the reduction in Dridex network activity was consciously effected by its operators, perhaps for operational reasons.

#### 5.4.1.2 Dorkbot

On a general note, the network activity of the Dorkbot operation appears to be varied and highly stochastic in nature, 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 than the other malware operations. This is indicated by the use of fewer unique URLs, domains, and IPs in the Dorkbot delivery operation. Further, I 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 ascertain the significance of this decline, as Dorkbot exhibits erratic levels of network activity, both before and after the takedown operation.

Analysing its network dynamics more closely, in Figures 5.2(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 observable with its use of IPs, as shown in Figure 5.2(d)–(e), albeit with a more pronounced, downward trend. It must be said that this pattern does not appear in Figure 5.2(c), which shows Dorkbot's (equally stochastic) use of domains gradually decaying for a few months before oscillating at a reduced level. Looking at its use of top e2LDs/IPs in Figure 5.2(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 server with 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 begins to utilise `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` (another 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 constant shifting in infrastructure 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 [12], 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 typically involves a single domain rotating between multiple IP addresses in a short period of time (i.e., within a single day).

Notwithstanding, particular Dorkbot domains that flux between several IPs per day were observed, such as `masterhossting|7772.in` and `superstar|7747.pw` (vertical bars inserted by author). 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 5.2(a) and 5.2(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 [12]. 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.

#### 5.4.1.3 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.

[3] `https://www.malwareurl.com/ns_listing.php?as=AS45945`

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 takedown is carried out against the Dyre malware operation. Specifically, when we look at Upatre's use of download URLs in Figures 5.2(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 [11, 5, 189]. 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 5.2(d). It is also interesting to note that, as we see in Figure 5.2(e), the Upatre operators ensure that their delivery servers are distributed among a number of countries. Clearly, the Upatre operation was being distributed through servers across multiple geographic regions, such as edge CDN servers for email services. A simple query of the data confirms this as I find Upatre malware being linked to hundreds of region-specific subdomains of various email servers in 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 [11, 5, 189], the Dyre malware heavily relied upon the Upatre dropper malware as its main infection vector. As such, it is plausible that the taking down of the Dyre operation could have had 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

[4] `https://whois.domaintools.com/afx.ms`

and the subsequent drop in Upatre downloads (which were predominantly through malicious email attachments), as the Dyre malware was only known to be a payload of Upatre. 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 5.2(d)–(e). On the other hand, as Figures 5.2(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 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 be indicative of 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. I find some evidence to support this hypothesis. First, in Figures 5.2(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 5.3(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, I identified

[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. $(r, \rho)$ as follows: Oct_Dec$(0.76, 0.60)$, Dec_Apr$(0.63, 0.45)$, Apr_Sep$(0.41, 0.11)$.

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. One must 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. Indeed, 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 a learned adaptation to previous takedowns.

### 5.4.2 Downloader Dynamics

In the last section, I analysed the network-level dynamics pertaining to each of the three malware delivery operations under study. In this section, I move my analysis on to the characteristics and download activities of the malicious binaries themselves, which are fundamental to malware delivery operations. In particular, I juxtapose the aggregate downloader dynamics, familial relationships (parent, children), delivery tactics, and polymorphic behaviours of the three malware operations simultaneously.

Figure 5.4 shows the aggregate download dynamics of each malware operation, while Figure 5.5 shows their relational dynamics (i.e., # of parent and child files), Figure 5.6 shows indicators of distributed delivery tactics, and Figure 5.7 indicators of polymorphic behaviour by the malicious binaries.

#### 5.4.2.1 Aggregate download activity

Figures 5.4(a)–(b) show the aggregate downloads and dropping activities of the Dridex, Dorkbot, and Upatre malware, whereas Figures 5.4(c)–(d) show the distributions of files that are downloaded $N+$ times, or drop $N+$ files. Immediately, we observe similar download behaviours between the Dridex and Upatre malware, but

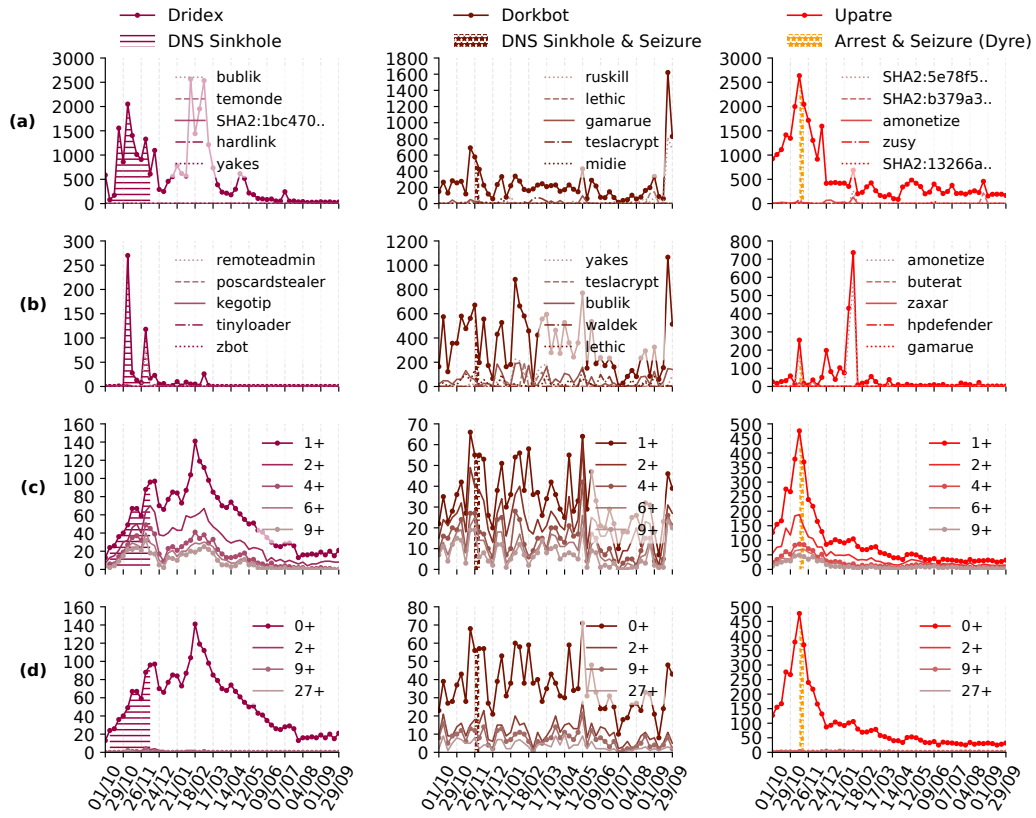


**Figure 5.4: Aggregate download activity:** **(a)** # of times downloaded; **(b)** # of drops by target malware; **(c)** # of SHA-2s downloaded $N+$ times; **(d)** # of SHA-2s that drop $N+$ files. Bursts of dropping activity by Dridex (during takedown) and Upatre (after takedown). Dorkbot activity 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).

significantly different behaviours from Dorkbot. This becomes a recurring theme in this analysis of download activities.

For the ***Dridex*** malware, we observe "bursts" of downloads and dropping activity during the takedown counter-operation, and 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 LEA 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, I identified a few instances of known malware families being delivered by Dridex, including some back-

door 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 that was known at the time to operate only as a payload rather than a dropper - suddenly engage in this practice of diversified, downstream malware delivery. Looking at Figure 5.4(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 5.4(d), almost none of the Dridex binaries engage in dropping activities. Rather, through querying the data, it was found 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, it is clear that specific strains of this malware were modified to drop other malware components onto victim systems.

With the ***Upatre*** malware, we observe similarities to that of the Dridex malware. As we see in Figure 5.4(a), and much like its network activity as analysed in the previous section, 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 5.4. In particular, of the files that the Upatre malware 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 (as opposed to other malware), such as convertad and amonetize, which are families known to bundle and and integrate with legitimate software.[6] This case study gives an indication of how convoluted file dependencies and delivery chains between malware, PUP, and benign software can be in the wild. As we look at the

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

bounded frequency plots of downloads per SHA-2 and drops per SHA-2 in Figures 5.4(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 the Upatre malware is known to operate mainly as a dropper malware. More generally, we find that the Upatre malware 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 5.4(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. I 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 5.4(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 $N = 1, 2, 3, ...$). This seems to indicate a weaker Pareto distribution (if any) in comparison to the other malware operations. The Dorkbot operation is also differentiated by its higher proportion of file SHA-2s that engage in dropping behaviour. Specifically, in Figure 5.4(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.

#### 5.4.2.2 Relational dynamics

In Figure 5.4 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 5.4(a)–(b) show the top 5 labelled software that either

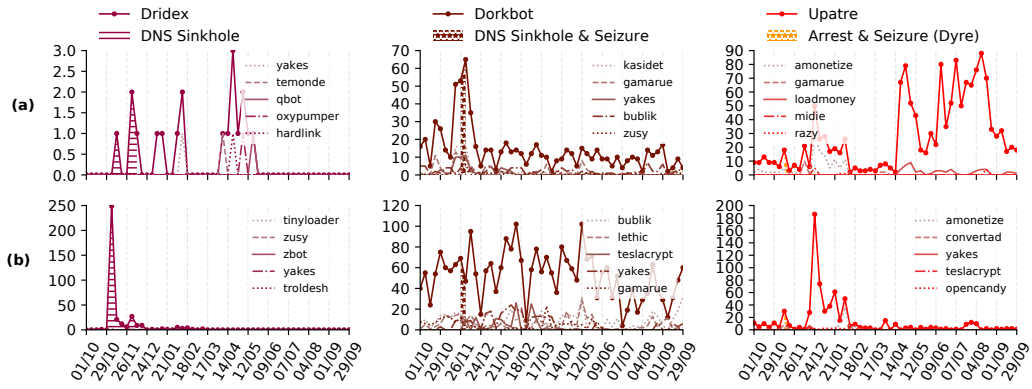


**Figure 5.5: Relational dynamics: (a)** # of SHA-2s that download target malware; and **(b)** # of SHA-2s dropped by target. N.B: the sharp increase in Upatre upstream droppers after mid-April, correlating with its increased use of DGA servers.

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 5.4(a) we see a sharp increase in `ruskill` downloads towards the very end of the observation window, while in Figure 5.4(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 5.5 shows the aggregate number of families involved in each malware operation. For the ***Dridex*** operation, Figure 5.5(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 [9].

The ***Dorkbot*** behaves very differently. As Figure 5.5(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.

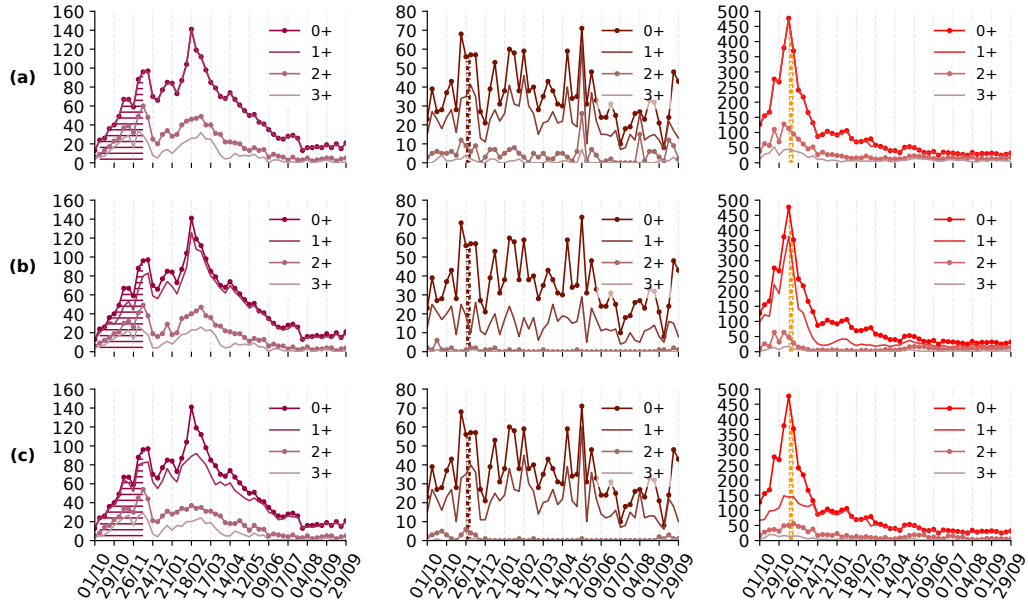


**Figure 5.6: Distributed delivery indicators:** **(a)** # of SHA-2s associated with $N+$ URLs; **(b)** # of SHA-2s associated with $N+$ e2LDs; and **(c)** # of SHA-2s associated with $N+$ IPs. Dorkbot downloads often without any traceable network resource, alluding to direct writing to filesystems.

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 5.5(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 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 number of upstream families, and (ii) the "top" families (e.g., `loadmoney`) accounting for only a small proportion of them. Though it is unclear what caused this change in behaviour, I note that it occurred from 14th April onwards – the same period Upatre began to use DGA download servers (see Section 5.4.1.3).

#### 5.4.2.3 Distributed delivery tactics

Figure 5.6 shows distributed delivery metrics of each malware operation: the numbers of SHA-2s (either being downloaded, or downloading other files) 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 5.6(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 victim's filesystem from the malicious process, as opposed to initiating the download from an external server. This is consistent with malware analysis reports, which identified spreading through USB flash drives as one of Dorkbot's infection vectors [13]. 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 5.6(b) shows the bounded frequency plots of ***e2LDs per SHA-2***, while Figure 5.6(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

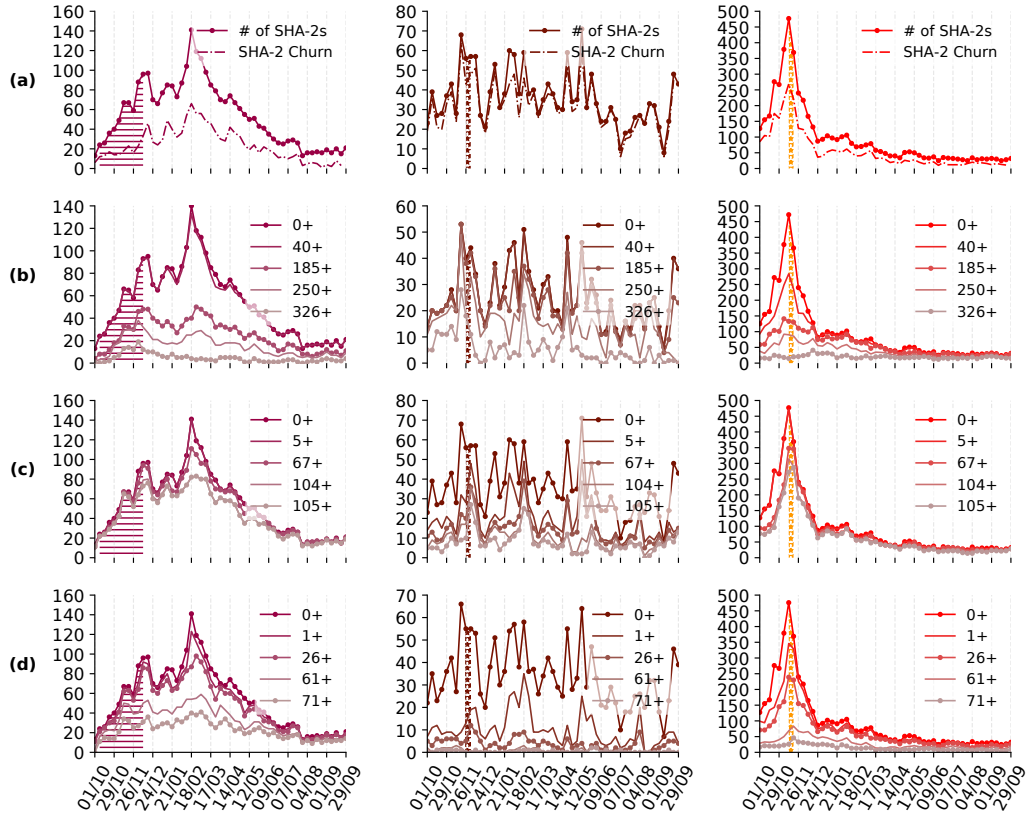


**Figure 5.7: Polymorphic characteristics: (a)** # of active SHA-2s and SHA-2 churn; **(b)** # of SHA-2s of size $N+$ KB; **(c)** # of SHA-2s with $M+$ malice score, where $0 \geq M \geq 128$; and **(d)** # of SHA-2s with $P+$ prevalence score, where $0 \geq P \geq 127$. N.B: malice and prevalence scores are assigned by Symantec security systems.

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 around the time of the takedown. 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 network component. This alludes to some binaries writing directly onto victim filesystems.

#### 5.4.2.4 Polymorphism

Figure 5.7 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 5.7(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 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 naturally 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 5.7(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 5.7(c) shows the distribution of assigned ***malice scores***, while Figure 5.7(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 gener-

[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. $(r, \rho)$ as follows: Dridex$(0.93, 0.93)$, Dorkbot$(0.74, 0.71)$, Upatre$(0.99, 0.93)$.

ate 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.

### 5.4.3 Summary of Results

In this chapter, I presented a comprehensive analysis of the activities of three different malware delivery operations, and how they evolved over a year in light of LEA efforts to disrupt them. A summary of these observations is presented in Table 5.2.

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

**Table 5.2:** Summary of LEA takedowns and observed behaviours of the targeted malware delivery operations.

## 5.5 Discussion

In this study, I conducted a detailed analysis of the dynamics and behaviours of three malware delivery operations over the course of a year. In this section, I take a step back to consider the implications of these findings. Specifically, I identify what security researchers and practitioners can learn from these observations, and how these findings could be used to adopt additional mitigation strategies. I also reflect on the limitations of this study, and opportunities for future work.

### 5.5.1 Lessons Learned

I uncovered a diversity of structural designs, behaviours, patterns, and responses to takedown attempts in the studied operations. The common themes that recurred in this study are as follows:

#### 5.5.1.1 Distributed delivery architectures

All three operations made significant use of distributed delivery infrastructures: Dridex used 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 [116, 137, 79, 194]. 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.

#### 5.5.1.2 Polymorphism and Pareto's principle

Polymorphism was rigorously employed by all three malware operations. 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. On the other hand, I laid bare manifestations of Pareto's principle across

all malware operations in that a minority of binaries were responsible for a majority of downloads or dropping activities. Although detecting polymorphic malware will be a continued challenge for the security community, this skewed distribution of activity towards a minority of malware binaries could point to efforts in detection being applied best in identifying these "super" binaries.

#### 5.5.1.3 Takedown resilience

Each malware operation responded differently and showed some degree of resilience to takedown attempts. 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 [188, 83]. 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 [186, 33, 88]. 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 [115] that the security community could seek to elicit more disruptive techniques from other fields of security research. For example, frameworks such as Situational Crime Prevention [61] could be adapted to derive countermeasures against botnet and malware delivery operations (see Table 6.4 in Section 6.4).

#### 5.5.1.4 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 [105]; (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 [85]; and (iii) *defiance* – an increase in offender activity in retaliation to an intervention, usually when the offender perceives the intervention as unjust or disproportionate [180]. Behaviours such as these are usually expected and taken into account in the application of interventions supported by environmental criminology. Similarly, in this study, I uncovered some interesting responses by 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. I 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, significant changes in the Upatre infrastructure only a few months after the Dyre takedown operation. 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 [59], implementing action-research models for botnet takedowns [115]); 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 such knowledge-sharing could be implemented.

#### 5.5.1.5 Unpredictable responses

At the same time, I also denuded 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, it was seen to engage 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). As such, this highlights the need for the security community to incorporate multiple data sources from different ecosystems to monitor botnet activity effectively. For instance, monitoring download traffic (as in this study) could be complemented and correlated 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, and, thus, how to use this increased knowledge to disrupt botnets more effectively.

### 5.5.2 Limitations

This work builds on the data and techniques used in a previous measurement study of the malicious file delivery ecosystem [116]. As such, the same data limitations apply, such as the limited view one has 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, I 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, I collected VirusTotal labels for a period of three years after the initial observations, maximising positive predictive capability. 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, I suspect such cases would be few given the reported accuracy of the classifier [177].

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 one is 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 one 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 one 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 one to observe *offender displacement* (i.e., a malicious operator replacing another's use of upstream delivery infrastructure).

## 5.6 Conclusion

In this study, I tracked and analysed three different malware delivery operations over the course of a year, studying the dynamics of their upstream servers and dropper networks. Through time series analysis, I studied the different facets of each operation and how they evolved over time in light of the law enforcement efforts to disrupt them. I 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. I 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.

# Chapter 6

# Bridging Information Security and Environmental Criminology Research to Better Mitigate Cybercrime

In this chapter, we take a step back to review the cyberthreat landscape as a whole and take stock of the countermeasures used to date in the hopes of deriving better and more holistic ones. To this end, I present a review of the cybercrime literature from two distinct perspectives: the information security perspective and the environmental criminology one. This work was undertaken as part of a collaborative project with *Dawes Centre for Future Crime at UCL*. A pre-print of this work is publicly available: *'Bridging Information Security and Environmental Criminology Research to Better Mitigate Cybercrime.'*

## 6.1 Introduction

Society and digital technology have become inseparable. The most developed countries are on the verge of true digitisation: the Internet of Things (IoT), driverless vehicles, and smart cities [221], while even in the poorest of societies, mobile technologies are becoming ubiquitous [23]. The Internet (or *cyberspace*) has been described as a 'real virtuality' [51]: an interactional environ-

ment that is rooted in the real world but transcends its spatial and temporal restrictions [219, 139, 203, 78, 209]. As a result, having a "digital identity", "going online", or "surfing the web" are no longer mere adages, but common, everyday realities. Now, communities and social networks can be globalised, giving us the ability to communicate in real-time and to meet in this virtual world. Indeed, our relationships, our activities, and our information are held in "cyberplaces", and not just cyberspace [210].

However, this entrenched influence that digital information now wields over society has also created new opportunities for the cybercriminal economy, which has already proven to be transnational, organised, and incredibly innovative. From the factory of spam and phishing emails [136, 188, 141, 134, 140] to meticulously planned romance scams [45, 211, 84, 112] and advanced-fee fraud [144, 103], identity theft, cyber fraud, and financial crimes are just some of the profitable avenues for the aspiring cybercriminal. Anonymous marketplaces and underground chatrooms are diversifying [80, 185], facilitating the trade of illegal goods and services (drugs, weapons, child sexual abuse images, etc) with the added benefits and protection of cryptocurrencies and escrow services [43, 146, 110]. Perhaps most devastatingly, such services have enabled the cybercrime economy to become increasingly organised, with cybercriminals regularly trading services with each other [183, 57, 104, 192]. The malware economy is just one manifestation of these rapid developments: growing from a "cottage industry" with a few, highly skilled individuals, to a massive and well-oiled criminal business, and constantly being refined, with wave after wave of new distribution vectors and attack patterns.

The interest in cybercrime prevention is profound. The information security field, which naturally has a strong technical focus, has been studying cybercrime and devising countermeasures since its inception. On the other hand, the environmental criminology field, which is multidisciplinary at heart and focused on traditional crime prevention based on traditional crime, has been comparatively slow in its reaction to cybercrime. There have been successes in crime prevention in these respective fields, but keeping up with cybercriminals has proven to be an arms

race. Furthermore, as I noted in Sections 1.2 and 2.7, little collaborative or interdisciplinary work between these two fields has been carried out. To keep up with cybercriminals, there is a pressing need for greater coordination and collaboration amongst computer security researchers and criminologists, among others, to better mitigate cybercrime.

In this study, I argue that combining contributions from information security and environmental criminology would benefit cybercrime research significantly, both to systematise past research efforts better and to identify promising future directions that draw from literature in both fields. To this end, I conduct a review of cybercrime literature from the perspectives of information security and environmental criminology, drawing parallels between these two distinct fields and eliciting how theories and frameworks from one map to research in the other. In this juxtaposition, I identify examples and opportunities for new cybercrime interventions by applying theoretical models from environmental criminology to information security research. These models also serve as a basis to help system designers evaluate whether their plans for defending the system have taken advantage of all techniques available, particularly for complex socio-technical systems which are poorly handled by existing IT-focused security standards. Finally, in arguing the need for a *new and complementary research direction* that combines contributions from information security and environmental criminology to mitigate cybercrime more effectively, I initiate this process in earnest: I discuss the concept of 'place' (amongst other core concepts relating to physical crime) and how I may define the analogous concept of 'cyberplace' for cybercrime. Ultimately, this would aid the transfer and adaptation of crime prevention frameworks to the context of cybercrime so as to provide a new outlook towards fighting it. My hope is that future thought and collaboration in this area would help the device of more effective cybercrime prevention strategies.

In summary, this study makes the following contributions:

- I present an overview of environmental criminology research and how the concepts presented in this area have been applied against cybercrime.

- I present a survey of cybercrime research from computer scientists, drawing parallels between the proposed mitigations and well established environmental criminology paradigms. To the best of my knowledge, this is the first study to draw from a wide range of literature and make these parallels explicit.

- I set the groundwork for future research directions that could see fruitful collaborations between the two areas. To this end, first, I propose some new cybercrime countermeasures using environmental criminology. This includes a framework for disrupting malware delivery and botnet operations using Situational crime prevention. Again, to the best of my knowledge, no such frameworks have ever before been propositioned.

- Second, I consider what 'place' means for cybercrime (amongst other fundamental concepts such as space-time, offender behaviours, and guardianship) and propose a new conceptualisation of 'cyberplace,' which combines three fundamental components: *location*, *state*, and *function*. I then present some motivating examples of how this concept could be used to derive new methods of cybercrime analysis and mitigations, including the facilitation of environmental criminology techniques for cybercrime. I argue that this concept could be applied and developed to identify cyberplaces (websites, services, software) that are at an elevated risk of attracting cybercriminal activity, and, thus, help system designers prioritise them better for preventative measures. This is the first study to define 'cyberplace' in this way for (but not limited to) the context of cybercrime.

The rest of the chapter is structured as follows. In Section 6.2, I will review the evolution of theories and practices in environmental criminology literature and how they may be applied to cybercrime. In Section 6.3, I will evaluate the core concepts of environmental criminology and how they translate to cyberspace and cybercrime. In Section 6.4, I will present an overview of the cyber threat landscape from the information security perspective. I will then evaluate their mitigations against cybercrime while highlighting some similarities in their approaches

with environmental criminology practices. I will end this section by considering some new, potential mitigations against cybercrime using environmental criminology. In Section 6.5, I propose a new conceptualisation of 'cyberplace' by (1) using an inductive approach to establish examples of 'place' contexts through a survey of cybercrimes (Section 6.5.1), and (2) considering what 'place' means in the real world (Section 6.5.2). I will then present some examples of how cyberplaces may later be classified in order to better analyse and mitigate cybercrime (Section 6.5.3). Finally, I will end the chapter with concluding remarks in Section 6.6.

## 6.2 The Evolution of Place-Based Theories and Practices in Environmental Criminology

In this section, I first explain why environmental criminology is an appropriate field of choice in furthering the security community's approach to mitigating cybercrime. I then assess the concept of 'place' in environmental criminology theories and practices and how its role has evolved over time. Finally, I juxtapose the applications of these theories and practices between physical crime and digital crime.

### 6.2.1 Why Environmental Criminology?

Before exploring the connections between environmental criminology, information security, and their perspectives on cybercrime, one may be considering at this point, "why environmental criminology?" That is, why should we consider this approach in dealing with cybercrime, in association with current information security efforts, and why not, for instance, other criminology subfields, or criminology as a whole?

Criminology is a broad and interdisciplinary field in the behavioural and social sciences, drawing primarily upon the research of sociologists, philosophers, psychologists, social anthropologists, biologists, and scholars of law. Criminology possesses an equally broad variety of theories towards understanding crime. Classical criminology emphasises on the sociological, anthropological, and biological factors affecting one's propensity towards committing crime. On the other hand, environmental criminology is a unique subfield in that, besides its application of the

scientific method to examine crime, it draws focus on the (previously overlooked) environmental and circumstantial factors that create criminal opportunity, rather than purely focusing on the individual characteristics alluding to the "criminal profile." In this regard, environmental criminology draws on research from a broader range of technical fields, from geography and economics to computer science and mathematics, to focus on and assess the proximal (rather than distal) aspects of a crime event that explain why it occurs, thus pointing to how it could be deterred. Environmental criminology, therefore, manages to elucidate how and why crime is not bounded to only those who "fit the criminal profile," but can be committed by any member of society, and why one may be found to commit a crime that others would deem contrary to their disposition or "character." Crime science [67] is an evolved field of environmental criminology with the principle focus on controlling crime and reducing "harm." Because of this relationship between the two fields, I will generally focus on the role of environmental criminology for the remainder of this paper, considering crime science as its natural symbiote.

Returning to criminology, and looking at the dimension of efficiency, classical criminology approaches lead to a heavy reliance on the judicial and penal systems to inhibit crime, through deterrence, punishment, and rehabilitation. Though such are likely necessary for society, they are, nonetheless, flawed as a sole solution. First, there is the attrition of justice: ever-diminishing proportions of offenders are successfully reported, then arrested, then indicted, then imprisoned, and then rehabilitated [94]. Therefore, the majority of offenders remain within or are reintroduced into wider society with little to no lasting, positive change. However, environmental criminology takes a pragmatic approach from the outset. With the primary focus on pre-empting and preventing crimes before they occur, this approach favours manipulating the immediate environment so as to deter one from committing a crime, such as by increasing the perceived risks or costs, reducing the perceived rewards or provocations, or removing the excuses associated with the commission of a crime [63]. In essence, this approach aims to reduce criminal opportunity for potential offenders.

Finally, the environmental criminological approach favours direct and practical methods, techniques, and technologies over purely theoretical discourse. Improvements in modern technology have resulted in the development of a number of crime analysis techniques (e.g., crime scripting [71], agent-based modelling [34], geographic profiling [171]), crime prevention techniques (e.g., situational crime prevention [63], crime prevention through environmental/urban design [121, 156]), and tools (e.g., digitised crime mapping systems and hot spot policing [82]). With this constant evolution alongside modern technology, it is no wonder that environmental criminologists have begun to shift their focus towards crimes committed using computer systems and the Internet. Given the similar focus held by information security researchers on mitigating cybercrime, it is equally unsurprising that these two fields could serve to complement each other in achieving this shared goal.

## Criticisms and Challenges

The field of environmental criminology does not come without its criticisms. One common and often argued criticism of the field is that crime interventions derived from it do not lead to an overall reduction in crime – they only cause crime to move elsewhere. This side-effect of crime intervention is a phenomenon known as *displacement*. However, most displacement research has found that displacement is far from an inevitable side-effect of crime intervention. Instead, crime interventions have been generally found to deliver net benefits with reduced crime [122].

More generally and as summarised by Wortley and Tilley [216], one of the main criticisms of environmental criminology is that it ignores the "root causes" of crime. As such, prevention efforts based on this approach are often characterised as only catching the "low-hanging fruit." Wortley and Tilley argue (as we will find in the review of the relevant theories and practices of environmental criminology) that human behaviour – and hence criminal behaviour – is by its very nature situational and intricately linked with the immediate environment. As such, the immediate situational and environmental aspects of a crime event are root causes of the crime themselves. With that being said, the proposition of environmental criminology is not to be a comprehensive solution to all crime – crime in itself is a highly complex

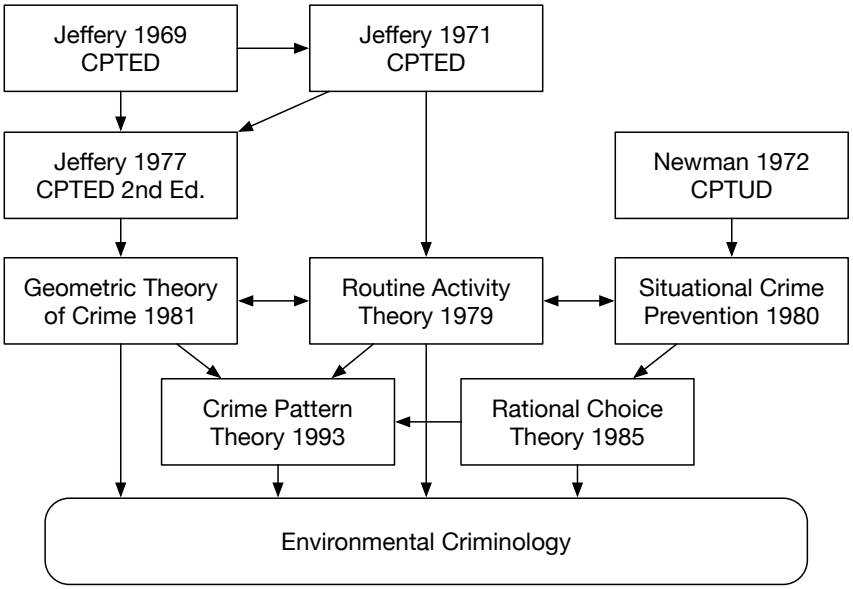


**Figure 6.1:** The evolution of environmental criminology.

phenomenon. Rather, as in its designation, it is designed to tackle the immediate environmental aspects that can lead to crime so as to control it.

One may also consider what benefit is there in investing in interdisciplinary research efforts between two fields that already share a similar focus (information security and environmental criminology). As I will show in later sections, the main benefit of a unified approach between the two fields, I believe, is the added structure and systematisation to the processes of devising, implementing, and monitoring cybersecurity mitigations. In essence, my argument is that the crime prevention theories and practices of environmental criminology may be extended to supplement the techniques already applied in information security.

### 6.2.2 Theories within Environmental Criminology

The foundation of environmental criminology (i.e., the study of crime, criminality, and victimisation) has a primary emphasis on the specific places and times where and when crime events occur. In environmental criminology, it is theorised that the characteristics of the immediate environment have a significant effect on whether a potential offender commits a crime or not. These environmental features are greater emphasised than other (distal) factors that are typically proposed within classical

criminology theory, such as the anthropological or neurological characteristics of "deviant" offenders.

I will review a condensed background of the concept of 'place' in environmental criminology, and how its role within this field has evolved over time. Figure 6.1 summarises this evolution of environmental criminology.

#### 6.2.2.1 Early studies

The idea that crime is non-uniformly distributed in space is not new. In fact, criminology research on these premises stretches back to almost 200 years, ranging from studies by Guerry [99], Quetelet [166], and Glyde [95] on early crime mapping and statistical applications to the social sciences, to works by Burgess [46] and Shaw and McKay [179] on the links between juvenile delinquency in urban areas and social disorganisation theory. A common thread within these works is the recognition of the fact that, with regards to areas and locations, crime does not occur uniformly. Instead, crime is heterogeneous in space.

Moving forward to the 1970s and 80s, environmental criminology called for a shift in focus on the specific places where crimes occur over other situational factors, such as the motivations of the offender, or the exploitability of the victims and/or targets of crime. Simply put, the "where" of crime was considered as or more important than the "who" or the "what" of crime. This led to the emergence of several fundamental environmental theories to explain crime, which, I believe, are better suited to mitigate new forms of crime, such as cybercrime.

#### 6.2.2.2 Crime prevention through environmental design

Jeffery [120] coined and formulated this term, often abbreviated as CPTED. The CPTED concept is simple: just as buildings and properties are designed to prevent damage from the forces of the elements, they should also be designed to deter and prevent crime. Such techniques include, for example, using a single, clearly notable point of entry to a private property to enable easy access control, or making communal areas highly visible to enable natural surveillance by its residents. Around the same time, Newman [156] developed Crime Prevention Through Urban Design (CPTUD) and the concept of *defensible space*: a model of residential environments

that exhibit territorial behaviours and senses of community to deter criminal activity within them. Ultimately, this model aims to provide perceptible cues to potential offenders that these areas are defensible (clearly bounded, regularly monitored, limited escape routes, territorial residents and users, etc) and, thus, unconducive to crime. Some argue that CPTUD could be interpreted as an application of CPTED through a subset of its dimensions [121, 26].

There are six key principles of CPTED, which, in varying degrees, are also relevant to cybercrime:

1. ***Territoriality:*** people tend to lay claim over an area that they have some form of ownership and will defend them against intrusion. Similarly, in the cyber context, people employ various methods of control over their computer systems, accounts, and websites to prevent unauthorised access (e.g., the use of passwords and security protocols) or to prevent malicious behaviours within them by enforcing terms of conditions for their users.

2. ***Surveillance*:** architectural designs that encourage residents to inhabit and interact with public spaces are more likely to deter criminal behaviour, such as by implementing increased lighting and unobstructed lines of sight. The concept of natural surveillance (as well as collective guardianship), is also useful in cyberspace. Particularly within social networks, forums, and e-commerce websites, members of these services can flag inappropriate or illegal content (and their users) for removal and can report software bugs in these services.

3. ***Target hardening:*** one may implement physical barriers (fences, gates, locks) to inhibit forced entry. In a digital sense, this maps to the use of authentication (password-protection, cryptography) and security technologies (antivirus, antimalware, firewalls, intrusion detection/prevention systems), and security personnel (network administrators, security analysts), which increases the difficulty for threat actors to compromise a service, system, or network.

4. ***Access control:*** one may deter criminal activity by defining site boundaries (fences, hedges), limiting access to a single point of entry or exit, implementing security systems and personnel, or guiding movement through a site. In cyberspace, this could involve the use of various authentication and security technologies as in target hardening, or utilising control flow integrity [20] and user experience (UX) design techniques.

5. ***Maintenance:*** a well-maintained site sends a signal to outsiders that people notice and care about what happens in the area, while those that are not well-maintained are more likely to entice vandalism, and, consequently, higher levels of crime ('Broken Windows' theory [213]). This principle is also relevant in the context of cybercrime. For instance, websites that are regularly monitored may be less desirable avenues for posting malicious or illegal content. Software that is regularly patched may be a more difficult target for attackers to exploit.

6. ***Activity support:*** providing clear signage on what are acceptable and unacceptable behaviours within a site can encourage expected patterns of use within it, e.g., including 'entrance' and 'exit' signs, or notices of criminal prosecution against malefactors. Likewise, shared services and websites that enforce terms and conditions (no hate speech, spam, or malicious hyperlinks, etc) are likely to encourage compliant behaviour (at least from real users) and discourage inappropriate ones.

CPTE/UD are a subset of the more general *Design Against Crime* (DAC) principles, as proposed by Poyner [164], which also includes crime prevention through product design [65]. On reflection, one may find that there are similarities between these crime prevention approaches and the security-focused design and maintenance procedures in systems security, such as the various levels of application security (secure coding, secure operating systems), system security (firewalls, antivirus, and antimalware software), and network security (intrusion detection/prevention systems, security information and event management systems). However, in CPTE/UD,

there is also a clear focus on empowering communities to deter criminal behaviour in their locales. Though one may argue that the use of computers is much more solitary than interacting in the real world, in actuality, as highlighted by other researchers [209, 210, 195], there is a great degree of online community in various forms, such as the shared use of computers and networks, software applications, forums, social networking, and e-commerce sites. Thus, these principles could be adapted to harness the power of online communities so as to prevent malicious behaviours within them.

#### 6.2.2.3 Situational crime prevention

Following the works of CPTE/UD [120, 156], and the successes of their implemented interventions, Clarke [60] argued for a situational approach to crime prevention. This approach is premised on offenders being rational actors, and that their choices and decisions towards crime are influenced by the characteristics of their immediate environment. For instance, a window regularly being left open could influence the commission of a burglary on that property, whereas a visibly secure property would be more likely to deter such a crime. Over time, and after some literary discourse [215], Cornish and Clarke updated the situational crime prevention (SCP) framework to establish 25 techniques [73]. The SCP framework can be summarised under five categories of techniques to deter potential offenders from initiating a crime event:

1. ***Increase the perceived effort.*** Physically, this includes the use of site security to deter offenders, while for cybercrime, this could involve automatically patching software and utilising application and network firewalls to deter hackers and malware.

2. ***Increase the perceived risks.*** Mitigations include implementing CCTV surveillance, or, for cybercrime, employing identification checks for money transfer services or de-anonymising cryptocurrency transactions.

3. ***Reduce the anticipated rewards.*** Examples of these mitigations include anti-theft mobile apps that can lock phones remotely. For mitigating cybercrime,

this includes using system backup policies to empower ransomware victims against complying to criminal demands, or using digital watermarking to detect piracy.

4. ***Reduce the provocations.*** This categority includes crowd control measures at venues to minimise stress and prevent altercations, or, in the digital sense, swiftly detecting and removing abusive and illicit content (which could breed further illegal activity) using automated filters and user reporting procedures.

5. ***Remove the excuses for crime.*** Examples include displaying roadside speed signs or using breathalysers in pubs, whereas for cybercrime, mitigations include displaying and enforcing stricter rules for social networking and e-commerce sites to discourage offensive and illegal behaviours.

Though this framework was developed predominantly for urban crime, it has shown to be useful in a wide variety of crime scenarios. For example, one adaptation has been made for counter-terrorism purposes [64], which maps the five types of SCP techniques across four necessary components of terrorism (targets, tools, weapons, facilitating conditions), generating a lattice of potential mitigations. Besides applying the 25 SCP techniques to various types of cybercrime, future research could go into developing customised frameworks for each type of cybercriminal operation, such as further adapting the counter-terrorism variant of SCP towards mitigating cyberterrorism (hacktivism, denial of service attacks) or malware delivery operations.

There are two potential effects of an intervention that are also the main criticisms of SCP [167, 93]: *crime displacement* and *crime adaptation*. Crime displacement involves the movement of crime (i.e., in space, time, modus operandi (MO), crime type, or the perpetrators and/or targets involved) as a direct result of a crime intervention. Cornish and Clarke [72] use rational choice theory to attempt to explain this phenomenon. Crime adaptation involves offenders learning of an intervention and adapting their techniques or MOs in order to bypass that intervention and commit the same crime. These crime phenomena could be of particular impor-

tance to cybercrime, as the ability of cybercriminals to move their operations elsewhere is effectively "free," in comparison to criminal operations in the real world, which can be more challenging. For instance, once a server being used for spam or other malicious activities is blacklisted, the operator could just change its IP address or domain name to circumvent this blacklist and continue their operations. Moreover, though there is a wide variance in their skill sets, some cybercriminals actively seek to beat the best cybersecurity defences and identify new ways to get around them (e.g., zero-day exploits, anti-analysis functionalities such as polymorphism and VM detection in malware). Therefore, it is even more important to devise interventions that are difficult to circumvent, or that would at least reduce the profits or increase the efforts and/or risks for cybercriminals who would do so.

#### 6.2.2.4 Routine activity theory

Cohen and Felson [68] proposed this theory as a macro-level explanation for crime rate changes in the United States between 1947 and 1974. This theory states that crime is less affected by (traditionally postulated) social causes, such as poverty, inequality, or unemployment, but more so by the immediate opportunity for one to commit a crime. In essence, they propose that "crime follows opportunity." That is, as more opportunities for crime arise, more crimes will occur. The core of this theory postulates that for (direct-contact predatory) crime to occur, three necessary components must physically converge in time and space: (i) a motivated offender, (ii) a suitable target, and (iii) the absence of a capable guardian (or some other controller, such as one who can handle the offender, or a place manager). In the real world, this could be exemplified by sexual assaults being more common at night as either the offender or the victim (or both) is more likely to be intoxicated and as there are fewer people out in public to deter them [77]. The same principle seems to apply in the cyber world. For example, studies have shown that botnet activities peak during the day and drop off at night [187], while most malware delivery is carried out on weekdays rather than weekends [116] – this is in line with when computers are used most often, which is usually for work. Cyberharassment and cyberbullying can only occur when the victims "come online" or access peer-to-


**Figure 6.2:** The 'crime triangle' of routine activity theory.

peer services (online forums, email services, social network sites, messenger apps). Hackers and their malware can only infiltrate a target system when it becomes available through a connecting medium (e.g., the Internet, a drive-by download on a website, downloading an email attachment, or accessing an infected USB device).

Over time, this simple but powerful trifactorial relationship has come to be known as the Crime Triangle (Figure 6.2), and is a key component of environmental criminology. Cohen and Felson also identify that most predatory crimes involve rational decision-making by the offender, particularly in qualifying "suitable" targets. With this premise in mind, they introduce the *VIVA* model (Value, Inertia, Visibility, and Accessibility) to explain how offenders qualify and select victims and targets, and how altering such dimensions could affect their perceived suitability for victimisation from the perspectives of these offenders. Alternative models have also been proposed, such as CRAVED (Concealable, Removable, Available, Value, Enjoyable, and Disposable), which is specifically designed for theft targets, or "hot products" [65]. As routine activity theory is core to environmental criminology, I will revisit the fundamental concepts of space and time, offender behaviours, guardianship, and how offenders identify suitable targets, and explore how these concepts apply in cyberspace and within cybercrime (Section 6.3).

#### 6.2.2.5 Geometric theory of crime

Building on routine activity theory, Brantingham and Brantingham [38] focus on the spatio-temporal relationship of crime. To elaborate, just as crime is non-uniformly distributed in space, it is also non-uniformly distributed in time. This theory ac-

counts for "peaks" and "troughs" in criminal activity across different geographical areas and places. The authors also identify a further three dimensions in understanding crime: the *legal dimension* (the creation, perception, and governing of laws); the *offender dimension* (the motivations of the offender and how they vary in time); and the *victim dimension* (why offenders select certain targets).

Going further, Brantingham and Brantingham [39] attempt to account for the non-uniformity and non-randomness of crime in their Geometric Theory of Crime. This work focuses on the urban landscape, theorising that an offender, just like non-offenders, will spend most of their time engaging in normal routines of non-criminal activity. It is through these routine activities that an offender develops their "personal awareness space", i.e., the areas and routes with which they are most familiar. It is theorised that when these awareness spaces intersect with the activity spaces of victims, it is in these areas that offenders conduct most of their criminal activity. The authors explain this theory by discretising the real world into (i) *nodes*: places which are central to the lives of people, and to and from which they travel (e.g., shops, schools, workplaces) – offenders tend to search for opportunities here; (ii) *paths*: routes that link nodes – people are often victimised along paths; and (iii) *edges*: physical and/or perceptual boundaries (e.g., rivers, major arterial roads) that separate distinguishably different areas – "outsiders" tend to commit crimes at these boundaries, while "insiders" tend to commit crimes within the bounded areas [41, 37].

The relationship between this theory and the activities in cyberspace is interesting. As I discuss further in Section 6.3.1, cyberspace is very different in its construction as compared to the real world: it is highly discretised (as opposed to the contiguity of physical space), with transitions between one online site to another being almost instantaneous. In this regard, the concept of victimisation occurring more along paths would need revision. However, it can be understood that online users form their own awareness spaces in cyberspace, based on the websites that they frequent and the services that they use. In turn, this familiarity may indeed reduce the perceived risks of using such services by these users (such as the con-

tinued use of illegal streaming and piracy sites), while also allowing cybercriminals to identify suitable attack vectors (e.g., vulnerabilities in a website) and targets (e.g., vulnerable users to socially engineer). This concept of awareness space is also comparable to the 'reconnaissance' stage of the Cyber Kill Chain [114] model, such as when crawler bots scrape websites for email addresses, or when malware scans nearby devices for vulnerabilities.

#### 6.2.2.6 Rational choice theory

For several years, rational decision-making had been a given assumption in modelling offender behaviours (e.g., situational crime prevention, routine activity theory). However, Clarke and Webb [62] formally proposed the Rational Choice Theory in 1985 to evaluate this proposition. By viewing different crime events in terms of (perceived) opportunity, costs, and benefits, this theory provides a possible explanation as to why and how offenders make rational decisions towards committing crime, as well as what may prevent such decisions from being made. This theory makes sense with physical crime, but perhaps even more so in the case of cybercrime. That is, there is a considerable rational element that may motivate one to commit a crime in the real world. However, other non-cognitive factors lead to a crime event, such as the immediate stresses, pressures, or prompting cues [215] that can, for example, spark an altercation in a bar, which may ultimately lead to an aggravated assault or even manslaughter. But, when we consider cybercrime, how these "situational precipitators" affect decision-making is not as clearly understood. Granted, the commission of a wide range of cybercrimes may require skill, motivation, and careful thought. For instance, it is probably far-fetched to identify a complex ransomware operation, spanning several weeks of activity, as a "spur-of-the-moment" assault.

Nonetheless, there are examples of cybercrimes that may be committed without prior intention, or even the knowledge that they are crimes. For example, one's participation in a heated online argument, or their reaction to an emotive piece of news, could quickly escalate to online abuse or cyberharassment. A user may stumble upon a bug in a website, and, instead of reporting it to the site owners, may

be tempted to see to what end exploiting that bug may lead, e.g., accessing accounts that had just had their credentials leaked in plaintext. Even hackers, though they may be considered as skilled and rational actors, may not always be aware of the criminality of some of their actions in cyberspace, thus, failing to distinguish between that which is lawful and that which is not.

#### 6.2.2.7 Crime pattern theory and repeat victimisation

The links between routine activity theory, the geometric theory of crime, and rational choice theory are apparent. Brantingham and Brantingham [40] attempted to leverage these links and synthesise these theories within Crime Pattern Theory. Through this theory, criminologists and crime scientists have been able to explain why crime occurs in certain areas, based on the intersecting activity spaces of offenders and their victims or targets. This theory classifies three types of crime hotspots [37]: *crime attractors*, which are places that are well-known to offenders for illegal activity and abundance in criminal opportunity, such as bars and nightclubs; *crime generators*, which are places that attract large crowds of people and where, being amongst them, offenders become aware of criminal opportunities there, such as shopping malls, schools, and entertainment venues; and *crime enablers*, which are places that facilitate crime due to their lack of place management, such as public parks and parking lots.

Repeat victimisation is a specific type of crime pattern, relating to the heightened risk of a victimised individual, demographic, property, or location, to being victimised again [89]. For example, the vast majority of homes remain unburgled while a minority of homes suffer multiple burglaries in a year. This increased risk of victimisation can be understood by the "flag" explanation, which relates to the characteristics of the victim or target that make them desirable to offenders (e.g., a home with broken locks or overgrown bushes blocking public visibility), or by the "boost" explanation, which relates to the role of repeat offenders in these crimes (e.g., a burglar identifying when a home is vacant, or learning techniques to overcome a security system) [161, 28]. The problem of repeat victimisation is that it is a concentration of a majority of crimes involving only a few victims and targets, and

being commissioned by a few offenders. As is the case with crime displacement, there are a number of factors that could cause this increased concentration of crime, such as geography (i.e., crime hotspots), types of (risky) locations (e.g., shopping malls, schools), the availability of targets (i.e., "hot products"), or the presence of repeat offenders or chronic victims. Besides the same people being repeatedly victimised, there are also *near-repeat victims*, who are different victims to the same (or similar) crime with some similarity to the initial victim (e.g., houses near to the one that was burgled are more likely to suffer a burglary than those further away). There are many examples of crime patterns also occurring in cyberspace (Table 6.1).

| Crime Pattern | Example |
|---|---|
| ***Repeat and Near-Repeat Victimisation*** | Cyberharassment against a victim and their close contacts [102, 168, 202].<br>Repeat sale and use of credit cards from Dark markets [104].<br>Repeat use of stolen accounts [158].<br>Malware spreading to nearby devices [147, 35].<br>Continued botnet activity: PPIs, spam operations, DDoS attacks,etc [33, 187, 192, 188].<br>Malvertisement drive-by downloads on same vulnerable browser plugins [222, 182, 91].<br>Outdated WordPress sites re-compromised twice as often as sites with up-to-date versions [206, 205]. |
| ***Crime Generators*** | E-commerce sites that enable buyer and seller fraud.<br>Online forums and game servers that enable fraud, sexual exploitation, and harassment.<br>Windows OS, which has 80% market share and fewer vulnerabilities than others but is the most attacked OS [8].<br>Websites with popular CMSes, which are more likely to be hacked or suffer fraud than others [184, 206, 205]. |
| ***Crime Attractors*** | Underground marketplaces such as Silk Road, which facilitate illegal products and services solicitation.<br>P2P software, pirate websites, and illegal streaming services are at an increased risk of delivering malware [198].<br>Browser extensions hosting malvertisements [222, 182, 91] and leading to other malware intrusions [127, 200].<br>Illegitimate app stores: large parts of the world can only access these though at higher risk to mobile malware [157]. |
| ***Crime Enablers*** | Social media sites with limited parental or speech-detecting guardianship enabling cyberharassment.<br>Websites and software with well-known vulnerabilities (SQLi, XSS, etc) and not regularly monitored or updated.<br>Devices and networks with no antivirus software or firewall protection attractive to malware.<br>Internet service providers with poor security hygiene attract malicious websites [190].<br>Criminals heavily use banks with poor operational security to handle their fraudulent payments [136]. |

**Table 6.1:** Examples of crime patterns in cyberspace.

| Crime Type | Theoretical Model/s | Intervention/Implication | Cybercrime Analogue/s |
|---|---|---|---|
| Number plate theft | Design against crime, VIVA, CRAVED | **Anti-theft plate [16]**: breaks upon removal (***inertia/removable***), rendering it useless for further criminal activities (***value***). | Spoofing attacks, online identity theft (e.g., email) |
| Terrorist attacks via station bins | Design against crime, VIVA, CRAVED | **Anti-terrorist rubbish bins [16]**: small bin mouth and volume prevents large disposals (***access***); translucent, frequently monitored, and can be X-rayed (***visible/concealable***); X-ray possible without triggering explosives, removable by a police robot (***value***). | Logic bombs, watering hole attacks |
| Supply chain crimes | Situational crime prevention | **Multiple techniques [100]**, e.g., protect ground floor warehouse windows by anti-ram posts (**increase effort**); screen consignments for prohibited articles (**increase risk**) | Malware delivery, phishing, spam, session hijacking, etc |
| Terrorism | Situational crime prevention | **Counter-terrorism SCP techniques and EVIL DONE [64]** target prediction framework (**E**xposed, **V**ital, **I**conic, **L**egitimate, **D**estructible, **O**ccupied, **N**ear, **E**asy) | Cyberterrorism/hacktivism: DoS, malware, phishing, etc |
| Identity theft | Routine activity theory | Identity theft victimisation [169]: 50% more likely for online banking and email/instant messaging users; 30% more likely for online shopping and/or downloading behaviours $\Rightarrow$ **focus on improving security for these services**. | Online identity theft |

**Table 6.2:** Examples of environmental criminology applied to crime problems, and their cybercrime analogues.

### 6.2.3 Practices of Environmental Criminology

I have already reviewed some theoretical models and how they can be used to analyse various crime types. In this section, I will cover some of the practical applications of these theories.

#### 6.2.3.1 Action research models

There are several systematic processes and risk management frameworks that have been used to implement crime prevention in a variety of public and private contexts (e.g., SARA [81], the 5Is [86], ISO 31000[1]). However, as other researchers [63, 100] describe, the common thread between these different approaches is that they are action research models, allowing researchers and practitioners to work together to:

1. analyse and define the problem, the ecosystem, and the relevant stakeholders (e.g., a shared computer, its programs, and its users),

2. analyse the situational conditions that permit or facilitate the crime event under study (e.g., infected app, malvertisement/drive-by, or socially engineered download),

3. identify, evaluate, and implement potential countermeasures (e.g., block third-party/untrusted apps by default, regular software updates, email "safe links", online safety reminders), and

4. assess the effects of these measures (e.g., diagnostics, user feedback), reiterating as necessary.

One can identify system vulnerabilities (such as for an OS environment, or a sociotechnical system) using the risk management variant of this approach, chiefly by identifying the parameters of a "good" system state, the goals and assets of the relevant stakeholders, and triggering events (criminal or mistaken) that counteract these goals as deviations from this state. Islam *et al*. [118] propose such a framework for reducing human-related risks in sociotechnical systems.

[1] `https://www.iso.org/iso-31000-risk-management.html`

Table 6.2 shows some examples of action research applied to real crime problems.

#### 6.2.3.2 Hotspot policing

Hotspot policing [82], which is based on crime pattern theory, involves constantly developing models of crime "hotspots" (points, streets, areas, chronic victims) and focusing law enforcement resources around them to efficiently deter crime. This approach makes sense, as focusing limited resources on the biggest sources of crime is likely to reap the most benefit overall. As I discuss later, this could also be applied to focusing resources on cyberplaces (websites, services, applications, etc) at an elevated risk of malicious activity (see Section 6.5).

#### 6.2.3.3 Geographic profiling

Geographic profiling [171] is an investigative technique to locate a serious offender's "anchor point" (e.g., their home or workplace), by connecting the locations of a series of crime events. This approach is similar to the clustering techniques developed by researchers [146, 101] that can (to some extent) de-anonymise the operators of Bitcoin transactions involving illegal activity, or the techniques that can be used to carry out traffic correlation [149] and de-anonymise Tor[2] users.

#### 6.2.3.4 Crime scripting

Crime scripting [71] is an analytical technique that is used to extrapolate the sequence of steps an offender may take to commit a criminal offence. For example, in a romance scam, fraudsters create a fake account on a dating service, they identify a suitable victim, they go through a grooming phase, followed by the actual fraud when the scammer asks their victim for money. Dissecting the various steps of an offence can be useful to better understand it and to identify potential interventions. The Cyber Kill Chain [114] is a crime script example that is already used for analysing system intrusions.

[2] `https://www.torproject.org/`

#### 6.2.3.5 Agent-based modelling

Agent-based modelling [34] (ABM) is a class of computational models that can be used to simulate an environment and assess how different actors interact with it and with each other. It is an analytical technique that is widely used in biology, sociology, computer science, and criminology. Researchers [35] have used ABM to model malware activities in heterogeneous environments.

## 6.3 Adapting Environmental Criminology Concepts for Cyberspace

As I have shown, the environment is a crucial element to crime in the physical world. For direct-contact crimes, it is imperative for a motivated offender and a victim or target to converge in space and time. The environment influences the offender's decision whether to commit a crime or not, their modus operandi, and whether such crimes are likely to be repeated. Therefore, altering the environment may be used to alter the decision-making of offenders, victims, or place managers, in order to prevent crime. These principles have already been applied in the real world, but it is of great interest to see how they may be extended into cyberspace and for dealing with cybercrime more effectively. In this section, I will consider the key concepts of environmental criminology, and how they may be operationalised in the cyber realm. Later, in Section 6.5, I will focus on the concept of 'place' in relation to cyberspace and propose how it could be adapted for cybercrime mitigation.

### 6.3.1 Space and Time

The spatial and temporal distributions of crime, which are based on the routine activities (or "rhythms") of law-abiding citizens and offenders alike, are the foundation of environmental criminological theory. In this regard, the dimensions of space and time are critical in understanding why, where, and when crime occurs. However, some aspects space and time differ significantly between the real world and the digital world, as do their effects on crime between these environments.

#### 6.3.1.1 The dimension of space

It is clear that the structure of cyberspace is considerably different to that of the real world. Whereas real space is continuous, the Internet is highly discretised with a node-edge topology (e.g., traversing webpages through hyperlinks). Cyberspace is also more ephemeral: websites can arise and disappear at rates much faster than land use in the real world. Yar [219] particularly notes arguments that cyberspace universally has "zero distance" between its points, hence making it difficult to meaningfully translate physical concepts, such as proximity and location, to the analogous problem of crime in cyberspace. Though there is indeed a theoretical basis for "zero distance" connectivity between computers, in reality, the concepts of proximity and location are still relevant in cyberspace for a number of reasons. First, cyberspace has a firm rooting in the physical world. The geographic locations of Internet service providers (ISPs), routers, and hosted web servers, and their relative connectivity, affect the structure of the Internet [203]. Political, economic, social, and cultural factors also affect the distribution of Internet infrastructure and usage. Numerous examples exist, ranging from the differing amounts of Internet activity and connectivity across different social demographics and different regions [219], to the intra- and international politico-economic factors that result in regional-specific restrictions on the Web (e.g., nationwide censorship of the Internet). Second, as Yar [219] notes, not all 'places' are equidistant when negotiating the Web. The ability of a user to find an entity on the Web greatly depends on how well other webpages reference that entity. Therefore, destinations that require many hyperlink clicks and numerous hops could be considered relatively distant from a given starting point, in comparison to destinations that require fewer hops. In this regard, the subjectivity of the user experience may be an important factor when considering distance in the cyber realm.

One can also deduce indicators of space and distance in cyberspace from the activities and the interactions of Internet users, both offenders and non-offenders alike (which I do in Section 6.5.1). Though the Internet allows for social networks to form irrespective of the physical distances between their members, one's physical

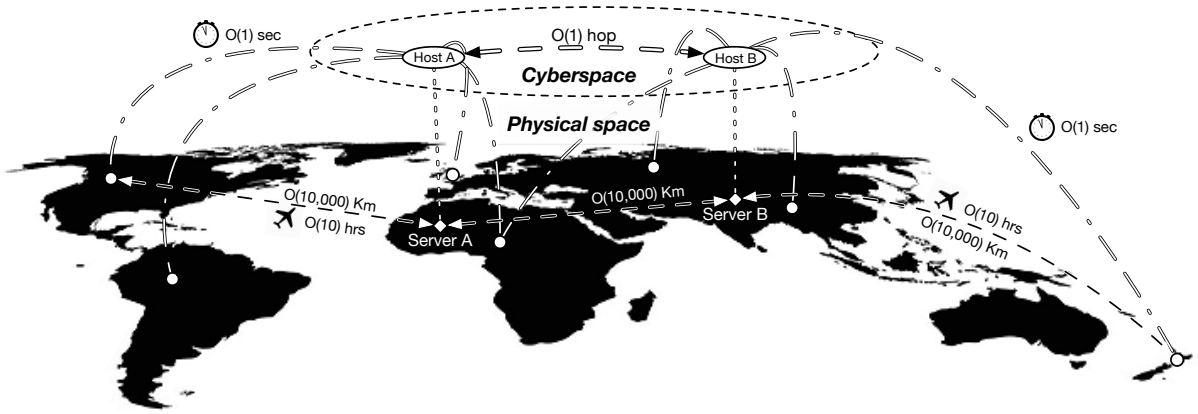


**Figure 6.3:** The contraction of distance and time in cyberspace.

proximities and relationships are likely to reflect in their online communications and activities [209, 210]. For example, one's email, social network, or phone contacts, or the peers in their local area network, reflect their real-world relationships – all of which may be exploited by, for instance, a motivated hacker or a cyberstalker. As suggested by the geometric theory of crime [41, 37], offenders and victims are more likely to establish awareness spaces around online services that they frequent (irrespective of the associated risks) and, thus, are more comfortable in using [139]. For example, users who visit pirate and infringing websites are less likely to install antivirus software, while, at the same time, being more exposed to malware [198].

When discussing space and cyberspace, another important topic is that of the jurisdiction differential between states and countries, which is not as clearly apparent in the digital world. That is, the laws and regulations of geographic regions enable some lines to be drawn on what constitutes acceptable behaviour and what criminal behaviour in the real world, as well as enabling the necessary agencies to enforce them. However, with the ephemeral and cross-regional nature of cyberspace, drawing and enforcing these legal lines is an ongoing challenge [96, 139].

It is worth noting at this point that little to no academic discourse regarding the juxtaposition between space and cyberspace (or 'places' and 'cyberplaces') and their relevance to cybercrime have ever been preceded by an exact definition of 'place' in the first instance. Perhaps, due to us being native to the physical world and not the digital world, we have taken the understanding of 'place' for granted and

deemed it too incongruent to be used in the cyber context without first addressing the ontological primitives of this concept, e.g., the three components of physical places: *location*, *locale*, and the *sense of place* [74]. I elaborate on this in Section 6.5.2.

#### 6.3.1.2 The dimension of time

What may first come to mind is the apparent instantaneity in the cause-and-effect of one's actions, such as the sending and receipt of an email message, or the rapid execution of a complicated task by a computer program. As Llinares and Johnson [139] (among others) note, this can make the subjective experience of time to appear shorter. However, the temporal dimension is still important in determining Internet usage, as there is a direct relationship between the real-world activities of people (especially at a micro level) and their activities on the Internet. For instance, users are more likely to access websites and download content for leisure and consumption outside of work hours [116]. Websites are more likely to be scheduled for maintenance late in the evening or during the early hours of the morning with respect to their local time zones. Major real-world events are quickly followed by online news and social media chatter.

The time factor is also relevant to the interaction between offenders (or their actions) and victims. For instance, users can only suffer threatening or abusive communications once they "come online" on a given service (e.g., social media, chat messenger, forum), or once they access their emails. Malvertisements and compromised webpages may remain dormant until a user accesses one of these pages before suffering a drive-by download attack. Victims of phishing attacks only become so after opening the malicious emails. Llinares and Johnson [139] note that, in general, peer-to-peer services can be characterised in one of two ways, based on the temporal mode of communication between Internet users. The first type can be characterised as *asynchronous* (store-and-forward or delayed) services, such as email programs, mobile SMS, social networking sites and apps that facilitate direct messaging (Twitter, Facebook, Whatsapp etc), and most static websites. The second type can be characterised as *synchronous* (real-time) services, such as VoIP services (Skype, FaceTime, Google Hangouts, etc) and online multiplayer games

that facilitate video, voice, and/or text-based chatting. The time between the action of a offender and the consequent effect on a cybercrime victim ranges from an instant to several months, or even years.

### 6.3.2 Offender Behaviours

Following rational choice theory [62] and its emphasis on offenders being rational decision-makers, there are countless instances of cybercriminals and their malicious agents making rational selections of when, where, and upon whom they commit their crimes. Intrusive cybercrimes are an example of the severe (IT-enabled) reduction in the costs and efforts associated with "travelling" and selecting suitable targets. For example, when hackers or their malware agents infiltrate victim systems or networks, they often scan for vulnerable machines or sift through the contacts of victim accounts for further targets [114]. This ability results in a multiplying effect on the potential damage that these crimes may cause. Furthermore, as IT enables users to easily connect with others on a global scale, cybercriminals may use this to their advantage in targeting as many victims as possible with the hopes of only successfully victimising a few in order to justify the effort. This approach is often the case with spam email, botnet, and ransomware operations, which enable large-scale attacks with a few successful ones generating most of the revenue [192].

In the real world, it has been shown that real or perceived anonymity may increase one's propensity to engage in antisocial behaviour in some circumstances (the Stanford Prison experiment [223] comes to mind). An interesting notion is the potential of anonymity in cyberspace to cause a similar increase in cybercriminal activity, though there is a present need for evidence-based studies to verify this. Cybercriminals have also been shown to be aware of the risks of detection concerning their operations. For instance, a large proportion of malware is capable of detecting antivirus software and honeypot environments (i.e., dynamic malware analysis VMs) and, consequently, suspend their activities in order to hinder security analysis [54, 31]. Malware also tends to utilise polymorphism (i.e., the same malware family appearing in different guises) in order to avoid detection [31]. Some compromised or malicious websites conduct visitor fingerprinting to show specific pages to

scrapers for search engine optimisation, malicious pages to end-users, and benign pages to potential security researchers and virtual machines [183]. Cybercriminals tend to use evasive and anonymising techniques to prevent detection, such as by applying fast-flux [109] and domain generation algorithms (DGA) [27] techniques as part of their botnet infrastructures, hosting and conducting illegal business on Tor onion services, or using anonymous cryptocurrencies to carry out financial transactions (e.g., collecting ransoms from compromised victims [125]), all in order to make it harder for law enforcement to catch those involved.

It is also interesting to consider how offenders become aware of criminal opportunities. As is the case in the real world, offenders are likely to form awareness spaces on the Web by way of the services that they regularly monitor or use. This characteristic would allow offenders to become aware of criminal opportunities, such as bugs in a website or software, or the (vulnerable) demographic of users for a given service. However, the Internet also affords offenders a higher level of surveillance to detect victims and criminal opportunity than in the real world, such as through the ability of a cyberstalker to observe when a user of a social networking app comes online, or a malware program to detect a working Internet connection on a victim's computer. Hutchins *et al*. [114] describe this more formally as the 'reconnaissance' stage of the cyber kill chain model.

### 6.3.3 Suitable Targets

Environmental criminologists have assessed the applicability of the VIVA framework (Value, Inertia, Visibility, Access) [68] to understanding how cybercriminals evaluate the suitability of potential targets [135, 219]. In summary, they find that the dimensions of 'value,' 'visibility,' and 'access' translate quite seamlessly from their physical interpretations to their digital ones. For example, focusing on 'value', a cybercriminal may assess the value of a target based on its financial potential, or its potential to increase the (notorious) reputation of the criminal. Information security studies reveal a strong rational element in how cybercriminals assess their targets. Paoli [76] explicitly draws this out in his analysis of timesharing security engineers in the 1960s and 70s who, like criminologists, conceptualise malicious users as ra-

tional actors who can assess the value of information. Thomas *et al.* [201] note that cybercriminals typically rent out compromised computers at varying prices depending on their regions, where computers from the West are usually more expensive in the cybercriminal economy than those from the rest of the world. Concerning the use of stolen email credentials, Onaolapo *et al.* [158] found that illicit users may ascertain the value of email accounts by executing searches using keywords such as 'bank' and 'money.' Turning to the other dimensions, the 'visibility' of a target from the perspective of a cybercriminal could translate to a victim's online presence, or the presence of well-known vulnerabilities in a service (e.g., a website bug, or a software CVE[3]). The 'accessibility' of a target may refer to a victim or system's attack surface by way of their software configuration (and associated vulnerabilities), or whether targeted data is stored within an access-controlled, digital environment or not. Even the aspect of 'inertia' – the difficulty associated with an offender's ability to transport a physical good, or to overpower a victim, due to their mass – is still relevant to cybercrime. Though some have considered it ill-conditioned for cybercrime because of the apparent "zero-mass" of digital data, recent discussions [219, 135] have shown that the sizes of target data, and the (inhibited) technical specifications of a cybercriminal's computer, may be forms of inertia that can influence cybercrimes such as information theft.

### 6.3.4 Guardianship and Natural Surveillance

Guardianship against crime is another relevant concept to the digital world. Principally, family members, neighbours, or friends may act as protectors in the physical world for would-be victims of crimes such as cyber harassment and bullying [36]. However, the concept of guardianship can be extended into cyberspace, both spatially (i.e., guardians can operate over the Web), and in an anthropomorphic sense (i.e., guardians may be software or 'bots' – not human beings alone). For instance, website owners, forum moderators, language filtering technologies can all act as guardians: to detect and prevent instances of cyber harassment and abuse. Popular social media sites such as Facebook, Twitter, and YouTube automatically detect

[3] `https://cve.mitre.org/`

and remove explicit content. Most Internet forums allow for the use of language filters to block inappropriate language. Likewise, the concept of natural surveillance [156] is also apparent in the cyber context, which talks of the ability of users to monitor spaces that they retain a shared interest. Notably, within social networking, blogging, and e-commerce websites, ordinary users can report inappropriate, rule-infringing, and/or illegal posts and adverts to the moderators of these services.

For more technically advanced crimes, capable guardianship and place management continue to be important to mitigating cybercrime. Website and software users are empowered to report bugs for their remediation – bugs which could otherwise be exploited and affect other users. Bug bounty programmes [90] create a financial incentive for such reports to be sent to the maintainer of the system and so allow the bug to be corrected. Network administrators, security analysts, and their myriad of security technologies (firewalls, intrusion detection/prevention systems, etc) are at the frontline of network-level protection, acting as guardians to users and devices within these networks. Endpoint- and application-level guardianship is also present through operating system, antivirus, antimalware, and web application technologies such as spam filters, unsafe site alerts, and web application firewalls, all employed for the protection of the end-user. In the mobile technology market, official app stores (e.g., Apple Appstore, Google Play) are more likely to vet third-party apps for malware and employ stricter development criteria than their unofficial counterparts [157]. However, despite the proliferation of digital guardianship, the task of improving system security has shown itself to be a continuous arms race between security practitioners and cybercriminals.

How and why some guardians (or "controllers" in general) are effective in deterring crime in the real-world, while others are not, is a topic that has only been investigated in recent years. In particular, one study [174] identified an important and defining relationship that could help denude this variance: the relationship between controllers and "super controllers" – those who regulate the incentives of controllers to prevent crime. No doubt, understanding the essence of this relationship towards preventing cybercrimes will be a valuable direction for future research.

## 6.4 The Cybercrime and Cybersafety Landscape

The field of environmental criminology has been, for the most part, primarily focused on crimes perpetrated in the physical world. On the other hand, the information security community has been studying the different facets of cybercrime and malicious computer activities for decades. Surprisingly, the parallels between the mitigations proposed by the information security communities and environmental criminology research have never been made explicit. The purposes of this section are threefold: (i) to give a general overview of the cybercrime and cybersafety landscape, and the current mitigations used; (ii) to draw parallels between the mitigations proposed by the information security community and the theoretical models of environmental criminology; and, finally, (iii) to present some examples of new, potential mitigations by applying environmental criminology (Tables 6.3 and 6.4), which are presented at the end of this section.

### 6.4.1 Anonymous Marketplaces

With the ongoing rise in malware distribution, widespread data breaches, and the unethical collection and use of personal data by various corporations and governments, there has been widespread attention and development towards privacy-enhancing technologies and regulations. One such technology that has become prominent is the Tor anonymous communication network. This encrypted network is resistant to common Internet tracking methods and enables users (who utilise it correctly) to effectively remain anonymous from all but the most technically capable adversaries. There are legitimate purposes for such a technology: users reading about sensitive topics, those with suppressed rights to freedom of expression, journalism, whistleblowing, or those who object to targeted advertising. Unfortunately, however, this anonymity has also been exploited to hide criminal activities, such as the trafficking of drugs, child sexual abuse images, violent pornography, and weapons. Even worse, underground forums and anonymous marketplaces (e.g., Silk Road) have arisen, enabling the convenient trade of such illicit products and services. Researchers have also observed the rise of 'crimeware-as-a-service' (CaaS) models [183] along with these "underground markets". These criminal business

models help to make cybercriminal operations (spam delivery, malware distribution, drug trafficking, money laundering) much more organised, automated, and accessible, especially for criminals with limited technical skills [192, 183, 57]. Such business models have been made possible because cybercriminals can network with each other on these underground services and exploit various outsourcing opportunities.

**Mitigations:** The primary methods of intervention towards illegal anonymous markets are server takedowns and arresting its operators. These approaches were seen in law enforcement's takedown of the infamous Silk Road marketplace in 2013, which, at the time, was nearly a monopoly. However, researchers have found that many more and diverse anonymous marketplaces have come to prominence since the takedown of Silk Road, with some (e.g., Silk Road 2.0) arising in less than a month. There is evidence of adaptation by these new marketplaces and their patrons, such as the increased use of encryption [185] and decentralised escrow services [110], and the diversification or specialisation in the types of products and services offered [80, 185]. These changes mirror the well-known criminological mechanisms of *crime displacement* (the net movement of crime elsewhere as a result of an intervention) and *crime adaption* (cybercriminals altering their operations in order to bypass an intervention), which are potential, undesirable side effects of some interventions.

### 6.4.2 Cryptocurrencies

Decentralised cryptocurrencies have gained significant traction over the past decade, with Bitcoin being the first and most widely used cryptocurrency. Bitcoin offers pseudonymity to its users, where accounts are not necessarily linked to real-world identities, but transaction details are publicly available in the distributed ledger. Other cryptocurrencies, such as Zcash, are designed for full anonymity [32]. Such properties are attractive to cybercriminals [43], making cryptocurrencies popular for illegal activities, like purchasing illicit goods and services [146], and enabling ransomware extortion [125], digital theft [173], and cryptocurrency laundering [44]. Kamps and Kleinberg [123] identified that cybercriminals take advantage

of the unregulated nature of some cryptocurrencies to engage in "pump-and-dump" schemes. This scheme is a type of fraud that involves three stages: accumulating a specific cryptocurrency coin, increasing its perceived value through misinformation (pumping), then selling it off to unsuspecting buyers at a premium price (dumping). **Mitigations:** Researchers such as Meiklejohn *et al*. [146] and Harlev *et al*. [101] have devised techniques that can, to some extent, de-anonymise the operators of Bitcoin transactions. Such techniques are especially useful for crime investigation and are similar to *geographic profiling* [82], which involves connecting locations in a series of crimes by an offender in order to locate their "anchor point" (e.g., their home). These are also practical implementations of the *'reducing anonymity'* situational crime prevention (SCP) technique, which increases the risks for cybercriminals by exposing their identities. With regards to pump-and-dump schemes, Kamps and Kleinberg [123] devise an anomaly detection technique in order to identify these schemes within time-series data of the trading prices and volumes of different cryptocurrencies. However, with an ever-increasing number of cryptocurrencies coming to the fore, and some that enable greater anonymity, it is clear that new approaches are needed to detect and discourage these sorts of criminal activities.

### 6.4.3 Cyberbullying and Online Abuse

With the advent of computer and networked technologies, the rapid adoption of the Internet has enhanced the abilities of end-users to perform their daily interactions – communicating, purchasing and selling products, exchanging information, working, and engaging in leisurely activities – without the limiting restrictions of time and space. Likewise, there has also been an increase in criminal opportunity through such technologies, thus enabling and (potentially) multiplying crimes that traditionally relied on physical, human-to-human interaction.

Studies have followed the physical-digital transition of such interpersonal crimes and antisocial behaviour, like cyberbullying [153, 202], cyberstalking and cyberharassment [212, 168], online hate speech [142, 107], and online child sexual exploitation and sexual harassment [102, 29]. These are only a few types of the crimes that have gained traction from such shifts in technology and society.

**Mitigations:** The default mechanisms for dealing with online abuse (in its many forms) typically involve reporting abusive or offensive content (and their authors) to the relevant service moderators (or *utilising place managers* from an SCP perspective). In extreme cases, such as the commission of violent threats, online sexual harassment, or child sexual abuse images, one may report such behaviours to the police. Although such actions can be useful, they are inherently reactive and often vulnerable to reporter biases (e.g., opinions of inappropriacy, cultural differences) or false reporting, and are probably less effective in preventing future occurrences [142]. Researchers such as Ioannou *et al.* [117], advocate the need for a proactive and multidisciplinary approach to dealing with online abuse. Even automated filters, which ought to blacklist hate speech and offensive language, are limited, as in they rely on predefined dictionaries of words. Such dictionaries are also inherently reactive and are inflexible towards misspellings and evolving language [178]. Consequently, researchers have developed some proactive techniques for mitigating these crimes.

Mariconti *et al.* [142] develop a supervised machine learning approach to automatically determine whether a YouTube video is likely to be "raided", i.e., to receive sudden bursts of hateful comments. Serra *et al.* [178] propose a text classification algorithm using class-based prediction errors in order to more effectively detect evolving and misspelt hate speech. Chatzakou *et al.* [53] develop a system that automatically detects bullying and aggressive behaviour on Twitter, using text, user, and network-based attributes. Founta *et al.* [92] present a holistic approach to automated abuse detection by supplying deep learning architectures with text and metadata-based inputs. Yiallourou *et al.* [220] devise a methodological approach that can be used to support the automated detection of images containing child-pornographic material. The successes of such surveillance strengthening techniques, which are indeed subsets of risk-increasing SCP techniques, are likely to increase the risk of getting caught for offenders and are just some of the multidisciplinary ways to deal with such problems. Of course, other forms of countermeasures exist. For example, the impersonation of minors by law enforcement

has been shown to be effective in apprehending offenders, while automated chatbots are being developed to profile potential offenders [29]. There is also the arrest and prosecution of the worst offenders [87]. Educating minors and Internet users to avoid online abuse victimisation is also an important, long-term initiative [214].

### 6.4.4 Cyber Fraud

Fiancial crime and fraud have also made a paradigm shift into the cyber world. The phenomenon of advance-fee fraud, or "419" scams (cybercriminals reaching out to potential victims with grandiose promises of wealth in exchange for advanced payments from them) have been well-documented by researchers [103]. Recent works have found such scams are more of a universal issue than once thought [144], rather than being one that only involves less economically developed countries. Cybercriminals have also been known to target other services for fraudulent activities, depending on their demographics of interest. For example, "419" scams are likely to be delivered en masse through spam email communications, where gullible recipients would self-identify themselves by responding to these emails [103]. Romance scammers are likely to operate on dating websites in order to manipulate emotionally vulnerable users [84, 112, 45]. Consumer fraudsters are likely to target large online marketplaces to commit buyer or seller fraud [204]. Various forms of identity fraud, facilitated through Internet-enabled theft of personally identifiably information (PII) (e.g., names, addresses, email addresses) or account credentials for common services (e.g., email, banking, social media) are also problems that the information security community closely monitor. Researchers have recognised that phishing emails and malware are common precursors to identity fraud [170], and they have monitored the illegal activities that subsequently ensue with such credentials [158].

**Mitigations:** The effective prosecution of scammers is necessary but often difficult due to the transnational nature of these operations and the relatively small amounts of money involved per fraud. Some engineering countermeasures are in use, such as the use of spam or phishing filters to prevent malicious messages reaching recipients, or blacklists that raise alerts or block known phishing websites. However,

the maintenance of such measures is a continual arms race, as cybercriminals are always adapting these spam messages or compromising new websites to avoid these blockers. It is possible for services to automatically detect scammer profiles, such as by their reuse of profile descriptions or profile photos [84]. On the other hand, perpetrators could also adapt to such countermeasures with ease. Arguably, the most effective countermeasures could be to reduce the profitability, or increase the required effort, for such crimes. An economic strategy, such as increasing the transaction fees or the necessary background checks for money transfer services, could be a set of mitigations that attack the profitability of such crimes. Awareness campaigns could also help to reduce the opportunity for victimisation, but perhaps more so if these campaigns are directed towards the most vulnerable, as identified by their personality types and victimisation statistics [45, 211]. With regards to environmental criminology, these are recognised as *market disrupting* and *target removing* SCP techniques, which involve reducing the rewards of crime by denying criminals the ability to steal, sell, or access a target.

### 6.4.5 Malware and Botnet Operations

One area of focus in the information security community is the study of malicious software, or *malware*, which is also the principal focus of this thesis. As I described in Section 1.1, the issue of malware came into prominence in the 1980s, but in recent times it has become a massive underground economy. In short, financial motivations (above others) have become a cornerstone to the design and proliferation of modern malware. Researchers have identified that modern strains typically carry a myriad of functions, no doubt for the purposes of monetisation. Malware families, such as Zeus [33], for example, can steal banking and financial credentials on compromised machines, log keystrokes and extract documents, or to encrypt victim computers to be held for ransom. Even worse, some malware families are designed to retain prolonged control of compromised devices and assimilate them into larger networks of infected machines, or *botnets*. These botnets may be used (or rented as-a-service) to facilitate distributed denial-of-service (DDoS) attacks against a target, to send spam emails [192], or to mine cryptocurrencies using the economic and

computational resources of the victims and their devices. Indeed, one the key observations presented in the previous chapter was the fact that malware can very often engage in completely unpredictable and undocumented behaviours, such as a banking trojan delivering other malware components and competing brands of banking malware to victim devices (see Section 5.5.1).

Malware distribution has been refined to infect as many viable victims as possible. Initially, there was a heavy reliance on human activity and manipulation, such as the need for victims to open spam email messages [187] or to be social engineered into activating a malicious file [155]. Nowadays, cybercriminals have developed distribution mechanisms to completely cut out the need for human interaction, such as delivering malware directly through automated browser-based attacks (or drive-by download attacks) via compromised websites or malvertisements [222, 152, 182]. To ease the lives of malware operators, the cybercrime ecosystem proceeded to come up with exploit kits – software packages that deliver a wide variety of exploits for different computer configurations [98]. This innovation, ultimately, increases the probability of a victim's system becoming compromised. In a further attempt to streamline malware delivery and lower the entry bar for cybercriminals, pay-per-install (PPI) schemes have also arisen in the cybercrime ecosystem [47]. These services are specialised botnets of infected devices that enable the distribution and download of new malware onto these already compromised machines. PPIs are set up and managed by a service provider, whom customers pay in order to infect machines with their own proprietary malware.

As described in Sections 1.1 and 1.2, the disruption of the malware distribution economy is an ongoing challenge. Cybercriminals increasingly implement new and numerous techniques in order to prevent their malware and botnets from being detected and disabled. Researchers have found that malware often obfuscates their outgoing communications, undergo polymorphism to "change their appearances", remain "silent" whenever they detect a possible malware analysis environment, copy themselves to multiple locations on a compromised machine, or distribute themselves over multiple devices on a network [31]. Botnet operators have

also been found to employ various tactics to avoid detection and takedown attempts of their infrastructures, such as implementing fast-flux techniques (the rapid rotation of IP addresses) [109], or domain generation algorithm techniques (the constant changing of domain names) [27], to hide the locations of their command and control servers.

**Mitigations:** The challenges of malware and botnet infrastructures are as complex as their operations. First, there is the issue of preventing malware infection and spread. Signature-based antivirus programs have long been the major defence in detecting and removing malware, along with intrusion detection systems and content filters. However, they struggle with the extensive manner of forms that malware now appears (polymorphic, metamorphic, compressed, encrypted, etc). This is compounded further by the fact that, according to some key findings presented in Section 5.5.1, most malicious activity is carried out by a tiny proportion of malware binaries, which means that the vast majority of malware signatures that are derived for detection may end up yielding little to no impact anyway. Antivirus programs that use heuristic methods for malware removal are now much more common (i.e., detection based on abnormal program behaviours). Notwithstanding, malware removal is still a reactive strategy, so proactive measures have also been developed. One such is the use of antimalware tools, which attempt to prevent malware attacks in the first instance through methods such as malware sandboxing, raising browser alerts on suspicious websites, and preventing the spread of malware if a device is infected. Another proactive approach is vulnerability assessment and management, which deals with providing regular system updates in order to remove known vulnerabilities. Such updates would reduce the success of drive-by download attacks, for example, thus minimising one's attack surface for malicious actors to exploit. Alternatively, and as identified in Section 4.4.1, benign online services that are being exploited by malicious actors to deliver malware could help to disrupt its spread by improving their security hygiene and practices. All these techniques are akin to *target hardening* that is applied in SCP and CPTE/UD frameworks, which aim to increase the difficulty of an attacker gaining access to their target. Although malware

delivery is not completely dependent on human error, this role is still substantial. Educating users to keep their systems up-to-date and on how to avoid social engineering attacks are some non-technical approaches that are also applied, such as with Action Fraud and their *#UrbanFraudMyths*[4].

Second, there is the issue of disrupting botnet operations. One important technique involves the infiltration of botnets by security researchers [33, 22, 55, 48]. Such techniques allow researchers to collect intelligence on cybercriminal operations, and identify weak points in their communication protocols for disruption, or locating their C&C servers for ISP takedowns. They may also be used to identify the owners of these botnets, such that law enforcement may arrest and prosecute them. However, with the estimates of Kaspersky Lab [2] indicating there could be hundreds of thousands of botnets in the wild, it is difficult to see the scalability of these techniques. More generally, as noted in Section 2.6, the success of takedown operations is mixed and highly dependent on the targeted botnet and other contextual factors. Alternatively, service providers may provide some mitigations. For example, email programs and social networking sites usually employ spam filters, which may consequently deter spam operations. However, these filters are often signature-based, so minor adjustments in the spam messages may cause them to go undetected. ISPs may use DNS sink-holing techniques and blacklists to prevent their customers from accessing sites known to be malicious. However, such techniques also come under the "arms race" issue of keeping up with the cybercriminals who constantly seek to evade detection of their infrastructures. Other economic measures are possible, such as pressuring ISP services to dissociate from "bullet-proof ISPs", which resist law enforcement and typically harbour these criminal activities, or pressuring financial institutions to dissociate from rogue banks, which liaise with cybercriminals, in order to effectively shut down their operations. Environmental criminology recognises these as *market disrupting* techniques (SCP), which aim to reduce the economic benefits of such operations until they are no longer viable.

[4]https://twitter.com/hashtag/urbanfraudmyths

Using the SCP framework, I provide a proof-of-concept matrix of potential countermeasures for disrupting malware operations in Table 6.4.

### 6.4.6 A Synergistic Approach

Though there are already clear parallels between the theoretical models of environmental criminology and the mitigative techniques proposed by information security, security researchers are yet to fully explore the *structured* analytical and actionable processes that environmental criminology has to offer. Firstly, past and current mitigations devised by security researchers only seem to represent or consider a subset of all the techniques that could be utilised, while simultaneously lacking a systematic approach to establish such techniques. Secondly, little attention seems to be directed towards the consideration, monitoring, and evaluation of the actual effects of interventions by security researchers, both with regards to the victims/targets and the malicious actors, and how they respond to these interventions. Ultimately, without considering the fulness of the crime prevention process, mitigations are more likely to fail (to different degrees) in controlling crime in both the short- and long-term, as cybercriminals may quickly identify alternative targets, crime types, or modus operandi.

| Cybercrime | Technique | New Countermeasure |
|---|---|---|
| Darkweb market solicitation | Situational Crime Prevention | *Increase the perceived risk* of incarceration for Darkweb market users by increasing the perceived number of undercover law enforcement agents within them and the awareness of their presence. Also, raise publicity of operations targeting low-level cybercriminals, as well as high-level ones.<br>*Increase the risks/Reduce the rewards:* introduce significant numbers of honeypot credit cards and email accounts under law enforcement control into underground circulation (dark forums/markets, surface web) |
| Cyberbullying and online abuse | Crime Prevention through Environmental Design | *Peer-to-peer monitoring*: users are periodically asked to review the anonymised messages (i.e., PII redacted) involving other users, who are either random users or members of the recipient's network, and confirming if these messages are rule-infringing or not (and why). This system could help to identify malicious communications and protect vulnerable users. |
| Romance scams | Crime scripting / Situational Crime Prevention | STEP 1: fraudsters create a fake account ⇒ *reduce anonymity:* require ID for registration; *utilise place managers:* identify and flag potential "predator" profiles<br>STEP 2: identify a suitable victim ⇒ *conceal targets:* identify vulnerable users (e.g., survey on registration) and offer increased protection (e.g., reduced visibility to unscrutinised users);<br>STEP 3: groom victims ⇒ *target harden:* educate users to identify common romance scam modus operandi (periodic awareness reminders and tests for users to identify scammers)<br>STEP 4: commit fraud by asking victim for money ⇒ *target harden* as above. |

**Table 6.3:** Some examples of cybercrime countermeasures using environmental criminology.

| Malware Value Chain | *Increase the perceived effort* | *Increase the perceived risks* | *Reduce the anticipated rewards* | *Remove the excuses for crime* |
|---|---|---|---|---|
| ***1. Malware development***<br>- malicious code-sharing<br>- outsourcing malware development<br>- engaging crimeware-as-a-service (CaaS) providers | - Disrupt dark markets and cybercriminal communication channels.<br>- Infiltrate dark markets. | - Publicise reverse-engineered malware and remediation.<br>- Publicise LEA operations in dark markets.<br>- Upscale malware analysis honeypots. | - Publicise reverse-engineered malware and resulting remediation. | - Publicise / educate public on the impacts of malware on victims.<br>- Increase reverse-engineering job roles.<br>- "Malware bounty hunting" programmes. |
| ***2. Malware delivery***<br>- Manual infections (social engineering, physical access)<br>- Automated infections (exploit kits, mass emails, network diffusion)<br>- Delivery networks (pay-per-install (PPI) networks, CaaS botnets) | - Educate public on cybersecurity.<br>- Harden targets (application, host, network, and Internet sec.).<br>- Improve ISP security practices (websites, CDNs, cloud hosting, broadband).<br>- Regulate ISPs and standardise security protocols.<br>- Regulate and police software advertisers and resellers (PPIs).<br>- Infiltrate PPIs and botnets.<br>- Takedown botnets (arrest, seizure, sinkhole).<br>- Pressurise "bulletproof" ISPs (C&Cs). | - Malware / exploit kit detection.<br>- Increase monitoring of CDNs and software marketplaces (e.g., GitHub, App Stores).<br>- Publish security advisories for malicious software.<br>- Less distinguishable malware honeypots.<br>- "Malware bounty hunting" programmes.<br>- Regulate and police software advertisers and resellers (PPIs).<br>- Infiltrate PPIs and botnets.<br>- Takedown botnets (arrest, seizure, sinkhole). | - Educate public on cybersecurity.<br>- Virtualisation / ephemeral sessions.<br>- Harden targets (application, host, network, and Internet sec.).<br>- Less distinguishable malware honeypots.<br>- Malware "vaccination" and disinfection campaigns.<br>- Regulate and police software advertisement and reselling industry (PPIs).<br>- Takedown botnets (arrest, seizure, sinkhole). | - Vulnerability bounty programmes (hosts, websites, web applications).<br>- "Malware bounty hunting" programmes.<br>- Publicise / educate public on the impacts of malware on victims. |
| ***3. Post-infection activities***<br>- Botnet control, CaaS<br>- Secondary cybercrimes (data theft, financial fraud, ransomware, replication, spam emails, crypto-mining, DDoS, PPI) | - Infiltrate botnets/CaaS services.<br>- Takedown botnets/CaaS services.<br>- Improve key service provider security and fraud detection (online banking, e-commerce, cloud services).<br>- Regulate key service providers and standardise security protocols.<br>- Implement DDoS prevention at ISP-level (broadband).<br>- Pressurise "bulletproof" ISPs (C&Cs). | - Infiltrate botnets/CaaS services.<br>- Takedown botnets/CaaS services.<br>- Monitor aggregate network/activity activity (service providers) for anomalies. | - Takedown botnets/CaaS services.<br>- Malware "vaccination" and disinfection campaigns.<br>- Virtualisation / ephemeral sessions.<br>- Harden targets (data encryption).<br>- Automated off-site backups.<br>- Educate public on cybersecurity.<br>- Improve key service provider security and fraud detection.<br>- Regulate key service providers and standardise security protocols.<br>- Implement DDoS prevention at ISP-level. | - Publicise / educate public on the impacts of malware on victims. |
| ***4. Post-malware activities***<br>- Consumption of stolen data<br>- Beneficiaries of damage<br>- Monetisation and laundering | - Disrupt dark markets.<br>- Regulate financial/crypto services.<br>- Pressurise shady financial services. | - | - Regulate financial/crypto services.<br>- Pressurise shady financial services. | - |

**Table 6.4:** A high-level malware value chain (left column) and a matrix of potential countermeasures using Situational Crime Prevention. N.B: the countermeasures proposed are not exhaustive – e.g., the *'reduce the provocations'* category was omitted.

## 6.5 Adapting the Concept of Place for Cyberspace

It is apparent that the environmental aspects and criminological principles that exist with physical crime can, to a large extent, be extended and applied to cybercrime. However, there is still the need for a clear conceptualisation of 'place' with respect to the digital space and cybercrime, since the concept of place is key to environmental criminology. In this section, I deduce what 'place' means in the context of cybercrime by examining various types of cybercrimes, and also by considering 'place' in the real world. I use the classifications of *cyber-enabled* crimes and *cyber-dependent* crimes to assess the concept of 'place' within each class of cybercrime, as defined by criminologists [145, 135].

### 6.5.1 Analysing Cyber-Enabled and Cyber-Dependent Crime Contexts

Cyber-enabled crimes [145] are crimes that occur in the real world but can be enhanced, expanded, and optimised by Internet technologies. That is, the Internet makes it is easier and cheaper for cybercriminals to find victims, to operate internationally, and to avoid getting caught. Some cyber-enabled crimes include identity theft, consumer fraud, various forms of cyber harassment and threatening communications, and the trafficking of illegal products or services through the Internet. Cyber-dependent crimes [145], on the other hand, refer to crimes that are only possible as a result of computer and networked technologies, such as hacking, malware infection, or botnet operations. Though these crimes will still have real-world consequences (e.g., identity theft, financial fraud), they can only occur through computers. Just as physical crimes occur in particular places, one can review a wide variety of cybercrimes and identify their associated 'cyberplaces,' especially focusing on the crime event and the context in which it is commissioned.

#### 6.5.1.1 Cyber-enabled crimes

Researchers have found that interpersonal crimes (cyber harassment, cyberbullying, cyberstalking, online sexual exploitation, etc) primarily occur on online services ranging from email, mobile phone, and instant messengers [212, 202] to blog sites,

social networking sites, forums, and chat rooms [168], to online games [102] and various VoIP technologies (i.e., real-time video chat) [29]. Financially-motivated cybercrimes have been found to occur also on these same online services, though the purposes and modus operandi of these crimes differ to interpersonal crimes. Whereas interpersonal crimes primarily involve the use of these services to emotionally traumatise and instil fear in victims, financial crimes, such as identity theft or consumer fraud, primarily involve the use of these services to induce potential victims for information and financial theft. For example, phishing scams [140] and advance-fee "419" scams [144, 103] typically occur on email services and web forums. Other scams that are unique to a type of online service, such as romance scams [84, 211] or buyer or seller fraud [204], almost always begin with the cybercriminal contacting the victim through these services (match.com, eBay, etc) and luring them into offline communications (email, telephone, or in-person), before defrauding their victims or committing other crimes. Profile name re-users, or "impersonators" – people who take advantage of reputable profile names because of their large followings – operate on social networking sites in order to engage their newly acquired followers in illegal activity (spam, illegal sales) [141].

Further still, crimes that involve the trafficking of illegal products and/or services (drugs, weapons, child sexual abuse images, malware, stolen credentials and credit cards, etc) also occur on online services, including through anonymised networks such as Tor, and using pseudonymous cryptocurrencies such as Bitcoin [52, 57] for conducting transactions. These services allow perpetrators to take advantage of the transnational nature of the Internet to upscale their distribution networks, their consumer markets, and, ultimately, their financial profits. Here, it is clear that the concept of 'place' is inherent: people know the sites to visit and the applications to use in order to carry out their online activities, whether legal or not.

#### 6.5.1.2 Cyber-dependent crimes

The nature of cyber-dependent (or "high-tech") crimes differ to that of interpersonal crimes. Primarily, cyber-dependent crimes exist only due to the presence of computers and the Internet, whereas interpersonal crimes (harassment, consumer fraud,

etc) already exist in the real world, but are only amplified through such technologies. More precisely, the primary target of high-tech crimes (hacking, malware, denial-of-service attacks, etc) are the computers and digital resources themselves, before the physical users who own or use them. Because of this, such crimes can, at times, bypass the need for human activity in order to occur. For example, security researchers have identified buffer overflow attacks (which occur on victim computers) as a common attack vector used by malware, such as the *Blaster* and *Code-Red* worm families [30, 147], completely bypassing the need for social engineering. Network protocol flaws have been exploited in order to establish malicious network connections, break into systems, and automatically spread malware throughout a victim's local network [126, 148].

Nonetheless, the "human element" is still exploited whenever possible. A significant proportion of malware is delivered to victim computers through social engineering, such as in the download of malicious email attachments. For example, the *Bagle.AH* and *Netsky.C* worms propagate themselves as email attachments to addresses that they find on infected computers, while also spreading through file-sharing peer-to-peer networks and local network drives [35]. Victims are also directed to malicious websites, via email, social media [141], or through malvertisements [182], where they consequently face bombardment by silent drive-by download attacks. Such attacks often result in the forced download of malware onto the victims' computers [152, 183]. The websites themselves that utilise vulnerable software are also targeted, such as through SQL injection and XSS attacks. In particular, websites that use content management systems, such as Wordpress, are more likely to be targeted and compromised than others due to their higher market share [206, 205]. Mobile users are not exempt from such devious strategies. For instance, security researchers have found that the *Mabir* worm spreads through bluetooth and MMS, prompting nearby potential victims to accept its installation [35], while the *Geinimi* trojan app is installed through third-party app stores, which consequently opens back doors on these devices and exfiltrates information [56].

There is also a relationship between the virtual places and geographic locations where cybercriminals tend to carry out their nefarious activities. For instance, the assimilation of compromised machines into botnets is common after malware exploitation. The prices of these 'bots' (hence their values) vary in underground markets based on the countries where they are located [201]. The command and control servers that are used to monitor and operate these botnets are typically hosted by "bulletproof" ISPs, which are based in countries resistant to law enforcement pressure and take-down attempts [183]. Similarly, malware that spreads through a host tends to look for peers that share some proximity to the victim, e.g., via email address books, shared computer networks, or social media connections. More generally, every computer and network resource is tied to physical infrastructure (e.g., computer memory, physical devices) that is located in the real-world, each presenting unique opportunities for crime.

In the context of these crimes, 'place' is more broad in scope, from websites and webpages to online services and web applications, to computer devices and their software, peripherals, and networks. These examples exhibit a marked difference in how 'place' is perceived between cyber-enabled and cyber-dependent crimes, where the former gravitate around online services and web applications (human-to-human activity), while the latter permeates every area of computing.

### 6.5.2 A Framework for Defining Cyberplace

The compatibility of environmental criminology with the practicalities of cyberspace merit, I believe, a new, complementary research direction towards mitigating cybercrime. Moving forward, the key to this new research direction is the development of a consistent definition of 'place' in cyberspace, or, simply, *cyberplace*. I have shown that the concept of place in cyberspace is strongly apparent on both the user- and device-levels: Internet users know the websites to visit and the applications to utilise for their work, leisure, or consumptive activities. Cybercriminals know the services to use to find and exploit their victims, to target assets, or to engage and trade with other cybercriminals. Routing devices know how and where to transmit information, through various network protocols (e.g., TCP/IP) in order

to reach any part of the world. Devices within a computer network are communicable by their own internal IP and MAC addresses. Even internally, every computer application, library, instruction, and chunk of data is addressed in memory and can, thus, be pinpointed to a physical device in the real world.

I have also shown that the way that each cyberplace is considered can be contextual, depending on the perspective of the actors who interact with them. For example, a web domain hosting a chatroom may constitute a single cyberplace for its users or a crawler bot, but in a networked sense (e.g., a router or a DNS resolver), it could represent different cyberplaces if it undergoes changes in its public IP address. Characterising these different cyberplaces would help the adaptation of crime prevention frameworks towards analysing and mitigating cybercrime more effectively.

**Revisiting the real-world.** Though I have spent considerable effort in identifying different cyberplaces through the contexts of cyber-enabled and cyber-dependent crimes, it is also worthwhile revisiting the primitive concept of 'place' from the real-world perspective. 'Place' has long been used and commonly understood within society for millenia, but it is only in the last few decades that geographers have conceptualised it as a particular location that has acquired a set of meanings and attachments [74]. It is recognised that place can be conceptualised in terms of the social interactions that they tie together, in that people go to places to engage in activities and that they interact with places themselves [143, 74]. However, more concretely, 'place' can be considered as a meaningful site that combines three fundamental components: *location*, *locale*, and *a sense of place* [74]. Location refers to the "where" or the absolute position of a place (e.g., the geographical coordinates of a university library). Locale refers to its material and tangible setting – the way that a place looks and what is contained therein (e.g., the university library on a busy street in central London, surrounded by buildings, having a gate and a courtyard, and long corridors and thousands of books within it). Finally, the sense of place refers to the abstract feelings and emotions associated with that place, which may be derived individually or by shared experiences (e.g., it is a place for study, there

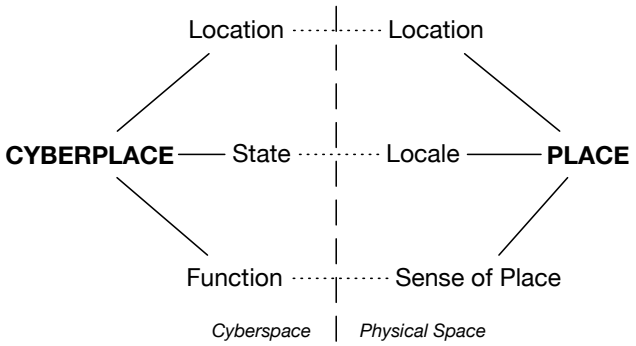


**Figure 6.4:** The analogous concepts of 'cyberplace' and 'place.'

is a presumed abundance of written materials, and there are personal experiences and history associated with the library). With these three elements, people (unconsciously) identify and differentiate places in the physical world. In a similar way, I propose that cyberplaces may be conceptualised as a combination of three fundamental components: *location*, *state*, and *function* (Figure 6.4), which I describe as follows:

#### 6.5.2.1 The **Locational** Component

This component is analogous to *location* with respect to physical places [74], which refers to the precise point or GPS location of a physical place. Thus, particularly in relation to how information flows within and between computers, the locational component of the cyberplace concept encapsulates information pertaining to the specific locations of cyberplaces in terms of digital address spaces. These address spaces encompass entire networks and computer hosts (e.g., IP and MAC addresses), peripheral devices, disk sectors, and even the memory cells within a computer. Therefore, just as computers and networks can inherently resolve the URL of a page on a website to a specific IP address and file directory, or resolve a function call in a software program to a specific address within a computer's address space, locational information enables us to specify the precise location of each cyberplace in cyberspace. In the same way, a change in "location" (address) will constitute a change in the overall cyberplace, which may be perceptible, such as would be in the case of a server changing its IP address, where the relevant DNS servers (the perceiving actors) would update their address mapping (A) records for

this domain and new IP. However, such changes may not be perceptible to actors who just access a server through its URL.

The locational component also explicitly links each cyberplace to a real-world location and device. For example, every website has an IP address, which can be traced to a physical server. Every instance of a file or an application can be traced to one or more memory locations within a computer system or peripheral device. Thus, this component provides useful information for the stage of implementing real-world interventions, such as pinpointing the real-world locations of cybercriminals and their servers for arrest and takedown. Of course, it is worth noting that the issue of criminals masquerading (or spoofing) their device locations is an operational issue, based on Internet design, and not a conceptual one. As I have previously discussed, 'location' can be an important factor from the criminal prospective (e.g., cybercriminals seeking targets from a certain country) and it is recognised by both human and machine actors (computers, malware, bots).

#### 6.5.2.2 The **Statal** Component

The second component of the cyberplace concept is the statal component, which encapsulates information relating to the state of a cyberplace and its tangible aspects at a specific point in time. This is analogous to the *locale* aspect of physical places [74], which refers to the way a place looks, its tangible aspects, and the surrounding environment. The tangible aspects of a cyberplace can be wide-ranging, and both internal and external to its environment. The internal state of a cyberplace (and how it is perceived from within) could be attributable to, for example, the content and hyperlinks on a webpage, or the webpage or software source code and the external resources that it utilises, or the files within a directory, or the overall design and appearance of the cyberplace. On the other hand, the extenal state of a cyberplace (and its visibility) could be assessed by, for example, the distribution of hyperlinks from external sites leading to a particular webpage (which affects the perceived distance from these sites to this cyberplace), or the popularity and generated traffic of a webpage (which may be captured with Alexa[5] or search engine

[5] `https://www.alexa.com/siteinfo`

rankings), or, in the case of a computer system, an application, or a process, whether a virtualised environment or the presence of antivirus, firewalls, or intrusion detection systems can be detected (e.g., a malware deciding whether to operate or hide its functionality).

The statal component is important because human and machine actors alike have some ability to assess the immediate context of a cyberplace (e.g., a webpage, a web application, or a desktop), and, thus, are influenced by those contextual factors in their decision-making. For instance, on a social media site, the appearance of some evocative content (e.g., a controversial image or post) may elicit abusive behaviours from some users, which would not have occurred otherwise. In fact, whenever there is a change on a website (e.g., a new blog post or comment), this site also changes state, and, thus, the potential activity that will occur thereafter. Such a change may, for example, affect the site's visibility on the Web (e.g., search engine ranking), raise its profile (e.g., Alexa ranking), or cause it to become a crime hotspot (e.g., a target for raiding). Likewise, the introduction of user input fields on a webpage (logins, search bars, contact forms – possible site vulnerabilities) may be followed by SQL injection and XSS attacks, before which such attacks would not have been possible. More generally, each time any online entity is updated, this entity changes state. Thus, if one version of this entity has a vulnerability that is targeted by cybercriminals, whereas an updated version does not, these two versions would represent separate cyberplaces from the perspective of the malicious actors. Therefore, users of unpatched versions of website and desktop software, for example, would be interacting with cyberplaces with elevated risks of victimisation (e.g., software exploitation, drive-by download). This component is relevant to both cyber-enabled and cyber-dependent crimes, as the states of websites and software are easily perceptible by both humans and machine actors, and could, therefore, influence their actions.

#### 6.5.2.3 The **Functional** Component

The third component of this cyberplace concept is the functional component, which encapsulates the intuitive function or purpose behind a cyberplace, to wit, why an

actor (human or machine) interacts with it. This is the cyberspace equivalent of the *sense of place* aspect of physical places [74], which refers to the feelings and emotions that are evoked when one considers a physical place (e.g., a restaurant) as well as one's mental expectations concerning such a place (e.g., the presence of tables, chairs, food and drink, and operating hours mainly within the afternoon or evening). As such, this is the most abstract component of cyberplace. However, it is also a vital one in that it provides the purposes to how and why actors interact (or expect to interact) with cyberplaces. For instance, the main reason for accessing news websites is to acquire local news information. News sites would probably be expected to categorise each news items with some (hyperlinked) headline, perhaps with some images or short synopses, and with some sort of order according to recency and/or significance (e.g., recent and important news at the top of the page, older and less important news at the bottom or archived). Generally, however, one would not expect to use a news website to purchase groceries, to buy a new laptop, or to access a remote server – one would expect to visit an e-commerce site for the former scenarios, or to use a remote-desktop application for the latter. In fact, it is probably for this reason that the Web is characterised using physical metaphors (sites, forums, chatrooms, email, desktops, etc) with which we are familiar in the real world. Furthermore, machine actors also share an innate understanding of cyberplaces in accordance to their creators. For instance, crawlers may be programmed to collect news items on a website with an inherent expectation of how those items are likely to be organised (e.g., the HTML tags to look for within the source code). Likewise, there are established protocols that enable computer systems to communicate with each other, depending on the services and ports involved (e.g., HTTP/S and ports 80/443 for web content, S/FTP and ports 22/21 for file transfers).

In regards to cybercrime (and more generally, the way that the Internet is used), the functional component of cyberplaces is important in that it is likely the defining factor as to how and why human and machine actors interact with cyberplaces, as well as to where, when, and by whom they are used. For example, as I discussed earlier, there is a general expectation that sites hosting illegal or explicit content

(streaming, pirate, or pornographic sites) are more likely to host malware than others. Yet, despite these risks, users are still drawn to these websites. Thus, the functional aspect of such cyberplaces could be what makes them desirable vectors for malware distribution.

### 6.5.3 Quantising Cyberplaces and Potential Applications

In a broad sense, I have defined the concept of cyberplace as a combination of three fundamental components. Of course, I can (and wish to) examine this concept in further discourse: the internal relationships between these components; its relationships with the real-world and cyberspace concepts of space, time, place, people, and machine actors; how it maps to various computer system and telecommunication models; etc. However, I will leave those discussions for future works and, for now, focus on how this concept may be applied in practice.

#### 6.5.3.1 Cyberplace classification

The three components of this cyberplace concept can be used to encapsulate all the information that is required to identify, describe, and differentiate cyberplaces on the Internet. Any cyberplace can be described in terms of its *function* (e.g., server, computer, website, web application, process, file, and each with their own types), its *location* (e.g., IP address, MAC address, file directory, memory address), and its *state*, which pertains to all tangible information relating to it at that point in time (e.g., webpage or software source code, types of content, included libraries, software versions, active processes, open ports). How one describes a cyberplace and to what level of detail would depend on the level of abstraction that is relevant to them and the underlying schema that they are applying. For example, focusing on the case of web content (accompanied by an understanding of Web structure), one could use functional information to coarsely categorise individual websites as cyberplaces, then to categorise individual webpages as smaller cyberplaces, then to categorise the individual features on these pages (search bar, blog entries, video player, etc) as even smaller ones, and so on and so forth. This would, in turn, warrant the inclusion of finer-grain locational information (e.g., from a domain and

IP address, to the subdirectory of a webpage, to a line in the source code) and statal information (e.g., from overall site attributes, such as the number of incoming or outgoing hyperlinks, or the server software version, to properties of individual pages or blocks of content, such as the number and content of images or HTML child elements, or the version of Javascript used). Clearly, there are an infinite number of ways in which this process may be implemented. Though such a classification could begin with manual efforts, a useful research direction may be to investigate the use of modern analytical techniques, such as through data science and machine learning, to heuristically extract the most efficient descriptors for cyberplace classification at each level of abstraction.

#### 6.5.3.2 Cyberplace risk modelling

The next step in cybercrime analysis involves classifying cyberplaces into various types of cybercrime hotspots (see Section 6.2.2.7) by applying the principles of crime pattern theory [37]. For example, one could label e-commerce and CMS-enabled sites, and computers using the most popular operating systems, as potential crime generators (i.e., they generate criminal opportunity due to the presence of many potential targets). On the other hand, sites that serve illegal content (pirate and streaming sites, underground marketplaces) could be labelled as potential crime attractors (i.e., they generate criminal opportunity as they are well-known for harbouring illegal activity). Finally, some social media sites and apps, and sites with poor cyber hygiene in general could be labelled as potential crime enablers (i.e., they generate criminal opportunity through lack of supervision or management). Either by empirical analysis, by generating probabilistic models, or by some other method, labelling cyberplaces as cybercrime hotspots could further guide their classification by cyber risk. Such classifications could express the likelihood of cybercriminal activity occurring at these cyberplaces, and, therefore, the risk of victimisation for their end-users and managers. One could even go further to derive expected user activities and potential cybercriminal modus operandi by way of analysing how users interact with these cyberplaces (e.g., UX analysis of websites and applications, control flow analysis of programs, vulnerability assessments,

penetration testing). Thus, these behavioural paths could be used as bases for establishing crime scripts for these cyberplaces (i.e., sequences of events, decisions, and actions that cybercriminals may follow preceding and following a crime event). Given that cyberspace is inherently *data-rich* and *discretised*, it is possible that these analytical models may be more suitable for this environment than the contiguous, real world.

#### 6.5.3.3 Cybercrime mitigation

Finally, these insights may be used for various mitigative strategies:

**Educating victims.** Awareness campaigns could be raised concerning the types of sites, services, and applications that pose the greatest risk of harm towards users or particular demographics, with the potential for new software to provide warnings before entering or using a high-risk site or application. In the same vein, providers of these cyberplaces (e.g., website owners, app developers) could be informed of their likely cyber risk level, and how they could minimise these risks.

**Altering cyberplace characteristics.** Another class of mitigations involves altering these cyberplaces to reduce criminal opportunity. In line with various situational crime prevention approaches, this could range from hardening these cyberplaces against vulnerability exploitation and improving security hygiene, to altering UX features and control flows of websites and applications in order to minimise malevolent or criminal opportunity (e.g., disabling or correcting vulnerable components that are likely to be exploited, requiring authentication in order to access a website).

**Applying tools and mitigation efforts more efficiently.** As research on repeat victimisation suggests, the Pareto principle applies in that a majority of crimes only involves a minority of offenders and victims. This suggests that the greatest reductions in cybercrime will arise by focusing cybersecurity tools and mitigative efforts towards educating and protecting the most exploited victims and targets, making the most crime-facilitating cyberplaces safer, and deterring, detecting, and apprehending the most prolific cybercriminals. Cyberplace classification techniques enable such efforts by identifying the cybercrime hotspots on the Internet, the cybercriminals who operate in or target them, and the users who are victimised there.

## 6.6 Conclusion

In this study, I conducted a review of cybercrime research from the perspectives of information security and environmental criminology. I presented an overview of how these two fields understand and (could) deal with cybercrime, identifying connections between their apparently disparate approaches. Upon review of a wide array of literature and cybercrime contexts, I provide motivating evidence as to why a new, complimentary research approach should be pursued involving these two fields. I initiate this process in earnest, first, by showing how frameworks from environmental criminology could be used to devise new cybercrime countermeasures – particularly for disrupting malware delivery and botnet operations; second, by proposing a conceptualisation of the immediate environmental contexts (or *cyberplaces*) where cybercrimes occur; and third, by providing some motivating examples of how the concept of cyberplaces, together with environmental criminology, could be used to better analyse and mitigate cybercrime. I hope that this work will encourage the wider research community to build upon this concept of cyberplace and its implementation in the transfer of crime prevention theories and frameworks between environmental criminology and information security. Above all, I hope that such collaborations will yield new and better approaches to cybercrime prevention.

# Chapter 7

# Conclusion

This thesis presented three major contributions towards measuring and disrupting malware delivery networks. In this concluding chapter, I summarise and reflect on these contributions to security research and practice, discussing the adopted research methodology and the relevance of the findings. I end this chapter with some concluding remarks, giving my own, personal take on the future of cybersecurity. I outline specific recommendations for future work in Chapter 8.

## 7.1 Research Scope and Contribution

This thesis comprises two main themes regarding malware delivery: measurement and disruption. In line with this, and as outlined in Section 1.3, the purpose of this research was summarised under two main objectives. In this section, I detail how these objectives were fulfilled, as well as the key findings and contributions that followed.

### Measuring Malware Delivery Networks

The first objective of this research was to measure malicious file delivery networks on the Web. More specifically, the following research questions were posed in a series of measurement studies:

**What does the malicious file delivery ecosystem look like?** In Chapter 4, it was observed that the malicious file delivery ecosystem could be partitioned into two distinct ecosystems: the first was a massive, PUP-dominated ecosystem consisting of an interconnected network of sites, server infrastructure, and unwanted files

(the Giant Component or GC, which was later termed the PUP Ecosystem). This ecosystem was responsible for over 80% of suspicious downloads on the Web and was temporally stable for at least a year. These findings led me to further investigate its structural backbone. The second ecosystem was a sparse one, comprising many, independent delivery networks, the majority of which were used to deliver malware (the Non-Giant Component or NGC, which was later termed the Malware Ecosystem). Despite the clear separation of these two ecosystems, it was not uncommon to find both types of unwanted software in either ecosystem, especially through mixed delivery infrastructures as exhibited in the Opencandy and Dyre case studies that I presented. These findings are particularly relevant to the security literature as they bring other relevant malware and PUP research into context. For instance, some researchers [127] had previously found malware and PUPs to be mostly disjoint problems, while others [131] found an overlap of 36.7% of droppers downloaded both malware and PUPs. These findings reported by other researchers sound contradictory at first, but my research shows how both are mutually inclusive. Going further, I quantified the relative proportions of PUP-to-malware in the wild, finding that PUP downloads dominate that of malware by a ratio of 17:2. To the best of my knowledge, I was the first to do so. Clearly, PUP is a more significant problem than first thought. This is particularly relevant given the surge of research interest in PUPs and their delivery networks in the last few years, which I discussed in Section 2.1.2.

**Are there differences between the network infrastructures used to download PUP and malware?** This question was addressed in Chapter 4 by comparing the network structures and delivery tactics employed in the PUP Ecosystem and the Malware Ecosystem. In general, it was found that the PUP Ecosystem exhibited a significantly higher use of domains with multiple IP addresses to deliver its files, which indicates the use of CDNs and (potentially) Fast Flux Service Networks. This points to PUP being delivered more commonly through well-known CDNs and online services that have servers in multiple regions (Google, MediaFire, Softonic). On the other hand, the Malware Ecosystem was found to deliver fewer types of files

(SHA-2s) per domain, indicating that these sites were not likely CDNs. More likely, these were malicious sites controlled by the cybercriminal operators, or benign sites that were compromised and being used to deliver malware. As I noted in the study, there are a number of possible explanations as to why there are structural differences between the two ecosystems. However, I could not confirm the specific causes for these phenomena without collecting additional data (e.g., DNS records, WHOIS records, site rankings). This is a suggested follow-up for further investigation.

**How do malicious file delivery infrastructures evolve over time?** In Chapter 4, I identified cyclic download patterns occurring each week, where infrastructures delivered more suspicious software during the weekdays than the weekends. I postulated that Routine Activity Theory [68] from environmental criminology could explain this observation, i.e., more infections occur when more users are online. Similar patterns had been observed by other researchers in other facets of botnet activity, such as when bots were most active, or when spam emails were sent [70, 186, 188]. Looking at the lifespans of delivery infrastructures, most were found to be short-lived, with only a minority of infrastructures being stable for a year. I posited that disrupting the most stable infrastructures (domains, IPs, files) would yield the most impact, as opposed to focusing on the ephemeral ones. These insights are particularly useful to stakeholders (researchers, analysts, engineers) who monitor or respond to malware-related activities on the Web.

**How do malicious operations respond to botnet takedowns?** In Chapter 5, I addressed this question by studying three malware operations that faced takedown attempts by law enforcement and security companies, and observing their post-takedown behaviours. Many interesting behaviours and activities (some of which were previously undocumented) were observed and noted in Table 5.2, while the key takeaways from these findings were discussed in Section 5.5.1. In summary, none of the malware operations studied were completely disabled after their respective takedown attempts, and each operation exhibited different responses thereafter in the short- and long-term. For instance, two operations exhibited immediate drops in activity after the takedowns, while one operation increased its activity during a

takedown attempt. At the same time, there were commonalities between the operations in the technologies they employed: the common use of distributed delivery servers, polymorphism, and takedown resilience technologies. There was also evidence of *spatial displacement* by some operators, where malicious operations moved from one set of infrastructures to another after the respective takedown attempts. There were also observations of significant changes in modus operandi by the botnet operators, such as when Upatre shifted to a DGA-based server architecture several months after its respective takedown attempt. I later categorised some of these behaviours as *predictable* (using environmental criminology theory) or *unpredictable*. My argument was that "predictable" behaviours could be pre-empted and factored into future takedown strategies, whereas "unpredictable" ones required further research to be better understood. These lessons give the security community deeper insight into how malware operators recover from takedown attempts, enabling more effective takedown strategies to be devised by law enforcement, security companies, and researchers in the future. It is worth noting that this research question is not particularly new: as I discussed in Section 2.6, other researchers have posed similar questions concerning other malware operations. However, one novel way this question was posed in this work was through the use of new analytical methods and datasets, which exposed malware behaviours that had never before been documented in security literature or industry reports.

## Disrupting Malware Delivery Networks

The second objective of my research was to identify better approaches to disrupting malware delivery networks. This was accomplished in two stages: first, by devising techniques to identify critical nodes in malicious delivery infrastructures that could serve as effective intervention points, and, second, by surveying the wider cybercrime literature to identify new methods of generating and evaluating countermeasures against malware delivery operations.

**Identifying effective intervention points.** I developed several methods for identifying critical nodes in malware delivery networks. In Chapter 4, I identified the core structural nodes (domains, IPs, files) of the PUP Ecosystem using a graph per-

colation technique. It was found that over two-thirds of these core servers were located in the US, indicating that ISP takedowns may be most effective in this region. Another technique was adopted to identify the most stable infrastructures in the malicious file delivery ecosystem. Using this technique, it was found that 26% of network infrastructures and 10% of file infrastructures were stable for a year. On the other hand, most infrastructures were found to be short-lived, with only 75% of network infrastructures being active for over 6 weeks. I argued that disrupting the most stable infrastructures would yield the most impact, as opposed to focusing on the ephemeral ones. Furthermore, in Chapter 5, using another technique to track entire malware delivery operations, I observed power-law dynamics in malware activity. Specifically, a minority of malware binaries were responsible for a majority of malicious downloads. Clearly, detecting these "super binaries" should be a priority for the security community. The most obvious beneficiaries of these methods are law enforcement, security researchers, and analysts who may use these techniques to track delivery operations and identify intervention points within them for takedown initiatives. Furthermore, documentation and source code for these analytical methods have been made publicly available[1].

**Surveying cybercrime literature for new mitigation strategies.** The key idea behind this survey was to go beyond the standard domain boundaries of cybersecurity research to consider how other fields could contribute to it. I focused on the potential contributions of environmental criminology, which the security community has only begun to consider in the last few years [191]. I also considered mitigations for other cybercrimes that intersect with the malware value chain (e.g., Dark market solicitation, cryptocurrency crimes, cyber fraud). There were several outputs of this survey. First, explicit parallels were drawn between information security and environmental criminology research, as well as areas that did not map as well. I achieved this by (i) providing an overview of environmental criminology research, and how some of its theories and practices have already been encapsulated in cybercrime research to date; (ii) exploring key, real-world concepts of environmental

[1] `https://github.com/ColinIfe/mdn`

criminology, and how they reflect in cybercrime and cyberspace; and (iii) presenting a survey of cybercrime research from computer scientists, again drawing parallels between the mitigations used and environmental criminology paradigms. Second, I demonstrated how environmental criminology frameworks and paradigms could be used to generate new cybercrime countermeasures. In particular, I discussed the use of action research models to monitor and manage the effects of a given cybercrime mitigation over time. I also proposed potential mitigations to a number of cybercrimes, the chief of which was a matrix of potential countermeasures to disrupt botnet and malware delivery operations (Table 6.4). Third, while reviewing the concept of *place* – a core concept to environmental criminology and studying crime in the real world – I proposed a new concept of *cyberplace* that could facilitate the use of environmental criminology models for crimes in cyberspace. To the best of my knowledge, these contributions to security literature are the first of their kind.

The knowledge generated from these studies benefit both the academic and non-academic communities, contributing to the body of knowledge for teaching and further research, as well as providing a synthesised knowledge base for stakeholders with an interest in cybercrime analysis and prevention. Such stakeholders include security specialists, sociotechnical system designers, and public policy practitioners. With regards to the novel concepts and proof-of-concept countermeasures that were proposed in this work, only time will tell of their utility in future academic research and security practice. Nonetheless, I believe that the security community and practitioners may apply these ideas, evaluate them, and build upon them.

## 7.2 Reflection

In this section, I reflect on my overall approach to conducting this research, the validity of the findings, and the limitations of this research.

**Measurement methodology.** In the measurement studies conducted, I used data that was sampled daily for a month, and one day a week for the remaining 11 months. This data was then represented using a graph abstraction, identifying IPs, FQDNs, URLs, and file SHA-2s involved in each download event. To enrich this

data, I used VirusTotal, AVClass, and National Software Reference Library ground truth data to establish which files were malware, PUP, benign, or unknown. I also used Autonomous System and IP geolocation data to identify the locations of the servers during the observation period. I then devised a number of heuristics and techniques to conduct specific types of analysis. Although the resulting studies were extensive, I do believe that there is untapped potential in the dataset used. In particular, as I discussed in the related work of Chapter 2, additional techniques and data enrichments could have been explored to identify other relevant phenomena in the malicious file delivery ecosystem. For instance, one could have collected DNS and WHOIS records for domains in the dataset to identify servers using Fast Flux and domain generation algorithms (DGAs). Alternatively, web crawlers may have been used to identify sites hosting exploit kits. With that being said, these additional measurements could be carried out in future works.

**Robustness and relevance.** One may question the robustness of the measurements presented in this thesis, both by way of the methodologies adopted and the age of the dataset used. It is important to reiterate that the data used and represented as graph networks only serve as proxies to server infrastructure and download activities in the real world. I have already discussed several limitations to the measurement methodologies, including external validity issues through the use of Symantec security data (Section 4.4.2), lack of ground truth and the effects of false positives (Sections 4.4.2 and 5.5.2), and difficulties in identifying file binaries that undergo polymorphism (Section 4.4.2). In my opinion, however, the most significant challenge to this kind of research relates to the technical issues that can arise when handling Big Data, particularly in identifying erroneous data and false positive results. In my experience, this was dealt with by questioning the validity of *every* finding and devising thorough tests and "sanity checks" to verify them. A clear example of this was the numerous experiments conducted to verify the discovery of a massive Giant Component in the malicious file delivery ecosystem (Section 4.3.1). Taking into account the extensive tests that were applied in this work, the numerous reviews of other academicians, and how the key findings are supported by other studies, I consider

the results presented in this thesis to be empirically robust. Nonetheless, for studies of this kind, one cannot completely rule out the possibility that some results may be slightly misleading or misinterpreted, simply because it is impractical to consider *all* confounding edge cases.

On the temporal robustness of these findings, I believe that the lessons learned from these measurement studies are still highly relevant today. This is despite the fact that the dataset was collected in 2015–16. This is because, firstly, as recent research and industry reports indicate, modern-day malware are still using the same delivery and evasive techniques described in these studies. Secondly, it is not uncommon for malware operations to last several years. For instance, some operations, such as Dorkbot or Dyre, were first discovered in the early 2010s, while modern-day malware, such as Emotet[2], also operated during the period of 2015–2016. In fact, some modern-day malware are evolved versions of the malware studied in this thesis (e.g., TrickBot[3] evolved from Dyre). Finally, a number of my findings re-echo what has been discovered in past studies that were carried out in different contexts. As such, these findings provide additional evidence to reinforce existing theories on malware delivery and the efficacy of takedown operations.

**Interdisciplinary approach.** In this thesis, I adopted an interdisciplinary approach to studying malware delivery. Specifically, I used a combination of systems security domain knowledge, data analytics techniques, and theories and concepts from environmental criminology to explain certain cybercrime phenomena. The clearest manifestation of this interdisciplinary approach came in the disruption portion of this thesis where I surveyed the cybercrime literature from the information security and environmental criminology perspectives. This study uncovered numerous parallels between the two fields, and provided guidance on how the security community could branch out with a unified approach to cybersecurity and cybercrime prevention. I began this process by demonstrating how frameworks from environmental criminology could be used to generate new cybercrime countermeasures with some proof-of-concept propositions. Going further, using domain knowledge from com-

[2] https://www.malwarebytes.com/emotet/

[3] https://securityintelligence.com/news/trickbot-malware-resurrects-the-ghost-of

puter science and geography, I proposed a new concept of *cyberplace* to establish situational contexts in cyberspace. Besides these contributions benefitting cybersecurity practitioners and adding to the academic literature, I hope other researchers, computer scientists, and engineers will be encouraged to adopt a similar, interdisciplinary mindset towards security research. Finally, as I noted when discussing the related work in Section 2.7, other studies using interdisciplinary approaches (particularly environmental criminology) to address cybersecurity challenges have come forth in the last few years [118, 59, 159]. I expect this trend to grow in the coming years, especially in relation to sociotechnical systems security.

## 7.3 Concluding Remarks

The problem of malware delivery is an interesting yet complex one – one which has no doubt been exacerbated by the rise of cybercrime networks and dark markets in recent years. In this thesis, I approached this problem from a high-level, data analytics perspective. Primarily, this was by virtue of a research partnership with Symantec from which I benefitted. In the latter stages, however, I delved into research with a more interdisciplinary outlook, both in relation to mitigating botnets and malware delivery operations, and mitigating other cybercrimes more generally. This latter research direction was borne both out of necessity and serendipity. Of necessity because, as I had discovered through my measurement studies, malware and PUP delivery networks were incredibly intertwined with various, benign web services, such as well-known CDNs and cloud hosting services – services which would require different remediation strategies to malicious ones. And, of serendipity in that the opportunity and timing to engage in this interdisciplinary research was purely fortuitous – thankfully, I was in the right place at the right time! With that being said, in concluding, I feel inclined to share some realisations I have acquired through this doctoral journey:

### People and Perspective in Cybersecurity

From my understanding, multidisciplinarity refers to people from different disciplines coming together to solve a common problem, while interdisciplinarity refers

to the integration and synthesis of knowledge and methods from different disciplines. In recent years, the need for different people, perspectives, knowledge domains, and skill-sets in cybersecurity and cybercrime prevention has been argued so often that it has almost become cliché. However, with my work in malware delivery, and with the work of the wider security community that I have come across over the past few years, I have only come to a fuller understanding and appreciation of this perspective. To me, both multidisciplinarity and interdisciplinarity will be key components to the future success of cybersecurity and cybercrime prevention. Take the analysis of malware delivery, for example:

Undeniably, low-level analysis of malicious file binaries will always be a cornerstone of malware delivery research: their characteristics, how they interact with different software and hardware configurations, how they exploit system vulnerabilities, and how they behave, communicate, and spread on local filesystems, over computer networks, and from external web servers. These phenomena are all pertinent to our understanding of malware delivery and to our ability to devise effective countermeasures against them. Already, one can see the need for a diversity of skill-sets for this level of analysis and intervention: knowledge and expertise in computer architectures, network protocols, reverse-engineering, provisioning sandboxes and analysis infrastructure, software and security engineering, detection systems, and so on.

However, the problem of malware delivery is not just isolated to individual computers and networks. Rather, it is a global one, spanning service providers and end-users across the entirety of the Internet, and culminating in a variegated and complex ecosystem of malicious distributors, payloads, and controlling actors. At this stage, high-level and large-scale studies (such as those conducted herein) become necessary to understand and mitigate malware delivery at a coarser granularity. As such, we begin to see the need for another group of skill-sets: primarily data analytics and data science, along with the related fields of statistics, machine learning, and big data.

But, going further, malware delivery is not driven by technical expertise alone – it is a criminal industry that is also driven by money, politics, and power, among other motivations. Cybercriminals communicate and network with each other. Dark markets allow criminal services to be exchanged and different parts of a criminal operation to be outsourced. Cybercriminals, being human, constantly make decisions to design, optimise, or alter their operations in light of new information and stimuli, such as a learned experience, or a LEA takedown attempt. This brings yet another dimension to an already complicated problem. Now, one must also consider the socioeconomic, psychological, legal, and regulatory ramifications of malware delivery activities, as well as the types of countermeasures that could be used to disrupt them.

Clearly, malware delivery – just one cybersecurity problem of many – cannot be solved by any single discipline. In almost perfect symmetry, malware delivery itself is not perpetrated by any one type of malicious actor or activity, but an organised conglomerate. Therefore, dealing with the problem of malware delivery more effectively *will* require the cooperation and coordination of different groups and different skill-sets: low-level and high-level computer analysts working together to derive more effective features to heuristically detect botnet activity and find better intervention points in malicious operations; high-level computer scientists working with crime scientists to derive more comprehensive analytical frameworks and mitigation strategies against entire malware operations and malicious delivery ecosystems; low-level computer scientists working with other disciplines to generate new approaches to systems security; and other stakeholders, such as Internet service providers, tech companies, security companies, law enforcement, and public bodies, taking a shared responsibility towards cybersecurity and cybercrime prevention.

Concluding, I hope the greatest contribution of this thesis is the increased *understanding* and *collaboration* of different researchers, practitioners, stakeholders, institutions, and communities in the fight to ensure cybersecurity is achieved.

# Chapter 8

# Extensions

In this final chapter, I outline my recommendations for future research based on the work conducted in this thesis.

## 8.1 Measuring the Malicious File Delivery Ecosystem on the Web

This study uncovered the structural characteristics of the malicious file delivery ecosystem on the Web, some high-level differences between malware and PUP delivery infrastructures, and aggregate retention rates and lifespans of delivery infrastructures. However, in the same vein, this study also unearthed additional interesting questions to be answered and challenges to be overcome regarding measuring malicious file delivery networks:

**Structures of malicious file delivery networks.** Interesting phenomena were identified at several points in the study, particularly when analysing the structure of the PUP Ecosystem in comparison with that of the Malware Ecosystem. In fact, towards the end of the snapshot analysis, a question was again raised on why the GC (PUP Ecosystem) exists. Because of data limitations, I could go no further than to proffer hypotheses for each observation (e.g., possibly higher CDN usage and/or use of Fast Flux in PUP Ecosystem, possibly higher use of evasive delivery techniques and/or compromised sites hosting exploit kits in Malware Ecosystem). As such, it may be worth investigating the structures of these ecosystems more deeply and testing the competing hypotheses generated in the first study. This extension could

be approached in a number of ways. One way could be to establish the types of web services present in each ecosystem through the use of additional domain metadata (e.g., WHOIS records, site traffic statistics, search engine data). Alternatively, or in complement, one could identify hidden connections between web services by querying historical site archives[1] for sites observed in this dataset, and matching the hyperlink destinations on each site with the sites observed in the same dataset. From a technical standpoint, it may be worthwhile exploring how one could incorporate Fast Flux and DGA detection into the proposed methodology (as discussed in Sections 2.3.1 and 2.3.2).

**Tracking delivery infrastructures.** A major part of this study involved tracking delivery infrastructures in time. The tracking technique devised for this purpose simply matched infrastructures between observation periods on a one-to-one and "best-effort" basis. Although this technique was sufficient for the purposes of this study (i.e., estimating the aggregate dynamics and lifespans of delivery infrastructures), more sophisticated tracking techniques would be needed to answer more pointed research questions. For instance, to consider and analyse more complex dynamics between delivery networks, one may consider the possibility of delivery networks splitting or coalescing over time, or groups of leaf nodes (binaries) that move from one upstream provider to another in lockstep. There are a number of other relevant topics that were not explored in this study that could be tackled in future works. For instance, one could extend this methodology to explore botnet activity detection based on the growth patterns of delivery networks. It may also be interesting to identify the business relationships that exist between different brands of unwanted software and upstream delivery networks, and how they change over time. With that being said, the methodology I devised for the second study in Chapter 5 could also be used to address such research questions.

[1] The Wayback Machine is a popular Internet archive: `https://archive.org/web/`

## 8.2 Tracing the Evolution of Malware Delivery Operations Targeted for Takedown

This study was scoped to measure the evolution of three malware delivery operations using network dynamic and downloader dynamic metrics. However, there are many opportunities to extend this methodology and conduct further research:

**Ecosystem dynamics.** This methodology helped us identify instances of a malware operation moving its operations from one set of infrastructures to another. This is an example of *spatial displacement* – a type of displacement from environmental criminology. However, following such a scenario, the methodology used could not allow us to determine the possibility of a second malware delivery operation taking the place of the first (i.e., making use of the upstream infrastructure or dropper network it abandoned). This would be an example of *offender displacement* – another behavioural phenomenon from environmental criminology. As such, extending the tracking and analysis methodology to detect such ecosystem dynamics around a given malware operation could be a worthwhile follow-up.

**Causality analysis.** Another important extension to this work is using causal inference to assessg the effects of takedowns on malware delivery operations. This is an interesting research direction because one could assess a number of causal relationships. For instance, considering the efficacy of takedowns on the targeted malware operation, one could assess the causal effect of takedowns on each aspect of a given malware operation (aggregate network dynamics, use of evasive techniques, download activity, presence of polymorphic malware). On the other hand, considering the shared infrastructure and business relationships between many different actors in the malicious file delivery ecosystem, one could also assess the causal effects of a takedown on other malware (and software) delivery operations that were not the intended targets (i.e., the side-effects of takedown operations).

**Large-scale analysis of software delivery dynamics.** Finally, as I alluded to in Section 5.3, it is possible to extend the methodology used in this study to analyse the activities of software delivery operations at scale (malware, PUP, benign). This could be helpful in investigating behaviours that are common or disparate between

unwanted software versus benignware. Likewise, one could further investigate the business relationships that exist between the myriad of software classes and families, and potentially identify software families that must be prioritised for intervention.

## 8.3 Bridging Information Security and Environmental Criminology Research to Better Mitigate Cybercrime

This study showed how the fields of information security and environmental criminology could work together to form a new, complementary research direction towards mitigating cybercrime. Furthermore, the contributions of this work serve as a platform for researchers and practitioners to test, assess, and refine the proposed mitigations. As such, I make the following recommendations for future work:

**Evidence-based cybersecurity.** Further research could be applied in assessing the proposed mitigations from this study, as well as new ones generated from the suggested crime prevention frameworks. In short, I recommend that the research community begins to apply the ideas of environmental criminology to generating more extensive cybercrime mitigations, applying them, and assessing them. This is of particular importance to security designers of complex systems (such as sociotechnical ones), who may often experience lapses in the security design process due to the many variables in such systems.

**Ontology of cyberplace.** The proposed concept of cyberplace is only the first stage in this research area – there is still a need to review and refine it. As I noted in the study, there is a particular need to investigate the ontology of cyberplace. That is, an investigation of how cyberplace relates with the real-world and cyber entities of space-time, place, human actors, machine actors (programs), data, computer systems and networks, etc. As stated, a clear understanding of cyberplace is a necessary precursor to understanding how real-world concepts and theories relating to physical places (e.g., "broken windows" theory) apply in cyberspace.

# Appendix A

# Additional Measurements of the Malicious File Delivery Ecosystem on the Web

## A.1 Lifespans of Delivery Infrastructures

In addition to computing the *lifespans* of delivery infrastructures tracked from October 1st, 2015 (Figure 4.13), the *presence* of the nodes within delivery infrastructures over a year was also measured. Figure A.1 shows a cumulative distribution frequency plot of the presence of different infrastructure nodes. This shows that, besides infrastructures that are observed one year on from October 1st, 2015, an

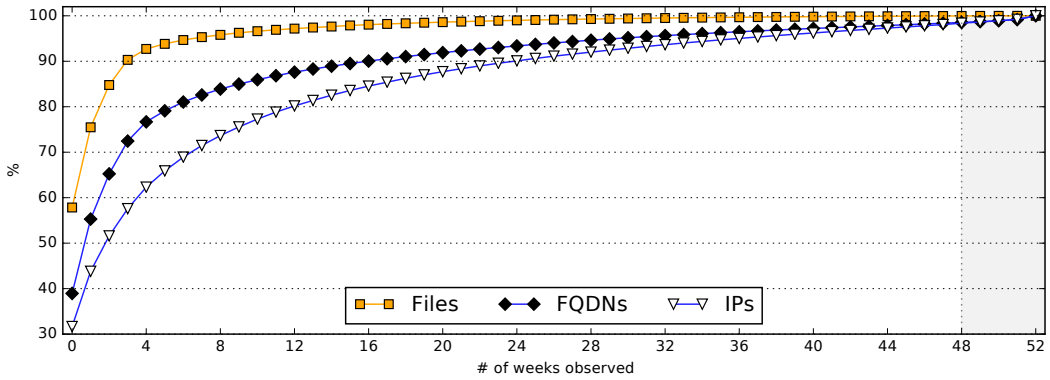


**Figure A.1:** Cumulative distribution frequency plot of the presence of different infrastructure nodes.

even smaller proportion of nodes ($< 3\%$) are stable for a continuous period of 48 weeks or more.

# Bibliography


[1] Adware/ExtCrome.syek. *https://www.avira.com/en/support-threats-summary/tid/143973/threat/Adware.ExtCrome.syek*.

[2] 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: 1-December-2018.

[3] 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].

[4] 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] 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].

[6] IPv6 martian and bogon filters. *https://6session.wordpress.com/2009/04/08/ipv6-martian-and-bogon-filters/*. [Accessed online: 24-May-2018].

[7] Sourceforge acquisition and future plans. https://sourceforge.net/blog/sourceforge-acquisition-and-future-plans/. [Accessed online: 4-November-2016].

[8] Top 50 products having highest number of cve security vulnerabilities. *https://www.cvedetails.com/top-50-products.php*. Accessed: 30-November-2018.

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

[10] VirusTotal. *https://www.virustotal.com*.

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

[12] 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].

[13] 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].

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

[15] Cloud power disrupts global malware (Dorkbot). *https://blogs.microsoft.com/on-the-issues/2015/12/17/cloud-power-disrupts-global-malware/*, Dec. 2015. [Accessed online: 11-September-2020].

[16] Design out crime: Case studies. *https://www.designcouncil.org.uk/resources/report/design-out-crime-case-studies*, 2015. [Accessed online: 2019-09-01].

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

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

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

[20] M. Abadi, M. Budiu, Ú. Erlingsson, and J. Ligatti. Control-flow integrity principles, implementations, and applications. *ACM Transactions on Information and System Security (TISSEC)*, 13(1):4, 2009.

[21] M. Abu Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A multifaceted approach to understanding the botnet phenomenon. In *Internet Measurement Conference (IMC)*, 2006.

[22] M. Abu Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A Multifaceted Approach to Understanding the Botnet Phenomenon. In *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*, IMC '06, pages 41–52, New York, NY, USA, 2006. ACM.

[23] J. C. Aker and I. M. Mbiti. Mobile phones and economic development in africa. *Journal of Economic Perspectives*, 24(3):207–32, 2010.

[24] I. Alabdulmohsin, Y. Han, Y. Shen, and X. Zhang. Content-agnostic malware detection in heterogeneous malicious distribution graph. In *Proceedings of*

*the 25th ACM International on Conference on Information and Knowledge Management*, pages 2395–2400. ACM, 2016.

[25] S. Alrwais, X. Liao, X. Mi, P. Wang, X. Wang, F. Qian, R. Beyah, and D. McCoy. Under the shadow of sunshine: Understanding and detecting bulletproof hosting on legitimate service provider networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 805–823. IEEE, 2017.

[26] M. A. Andresen. *Environmental Criminology : Evolution, Theory, and Practice*. Routledge, Mar. 2014.

[27] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon. From throw-away traffic to bots: detecting the rise of DGA-based malware. In *USENIX Security Symposium*, 2012.

[28] J. Ashton, I. Brown, B. Senior, and K. Pease. Repeat victimisation: offenders accounts. 1998.

[29] K. V. AÇAR. Webcam Child Prostitution: An Exploration Of Current And Futuristic Methods Of Detection. Apr. 2017.

[30] M. Bailey, E. Cooke, F. Jahanian, and D. Watson. The Blaster Worm: Then and Now. *IEEE Security Privacy*, 3(4):26–31, July 2005.

[31] U. Bayer, I. Habibi, D. Balzarotti, E. Kirda, and C. Kruegel. A view on current malware behaviors. In *LEET*, 2009.

[32] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *IEEE Symposium on Security and Privacy (SP)*. IEEE, 2014.

[33] H. Binsalleeh, T. Ormerod, A. Boukhtouta, P. Sinha, A. Youssef, M. Debbabi, and L. Wang. On the analysis of the zeus botnet crimeware toolkit. In *Privacy Security and Trust (PST)*, 2010.

[34] E. Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the national academy of sciences*, 99(suppl 3):7280–7287, 2002.

[35] A. Bose and K. G. Shin. Agent-based modeling of malware dynamics in heterogeneous environments. *Security and Communication Networks*, 6(12):1576–1589, 2013.

[36] A. M. Bossler and T. J. Holt. On-line activities, guardianship, and malware infection: An examination of routine activities theory. *International Journal of Cyber Criminology*, 3(1), 2009.

[37] P. Brantingham and P. Brantingham. Criminality of place. *European journal on criminal policy and research*, 3(3):5–26, 1995.

[38] P. J. Brantingham and P. L. Brantingham. Introduction: The dimensions of crime. *Environmental criminology*, pages 7–26, 1981.

[39] P. J. Brantingham and P. L. Brantingham. Notes on the geometry of crime. *Environmental criminology*, pages 27–54, 1981.

[40] P. L. Brantingham and P. J. Brantingham. Environment, routine and situation: Toward a pattern theory of crime. *Advances in criminological theory*, 5(2):259–94, 1993.

[41] P. L. Brantingham and P. J. Brantingham. Nodes, paths and edges: Considerations on the complexity of crime and the physical environment. *Journal of Environmental Psychology*, 13(1):3–28, 1993.

[42] T. Brewster. 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].

[43] S. D. Brown. Cryptocurrency and criminality: The bitcoin opportunity. *The Police Journal*, 89(4):327–339, 2016.

[44] D. Bryans. Bitcoin and money laundering: mining for an effective solution. *Ind. LJ*, 89:441, 2014.

[45] T. Buchanan and M. T. Whitty. The online dating romance scam: causes and consequences of victimhood. *Psychology, Crime & Law*, 20(3):261–283, Mar. 2014.

[46] E. W. Burgess. Juvenile delinquency in a small city. *Journal of the American institute of criminal law and criminology*, 6(5):724–728, 1916.

[47] J. Caballero, C. Grier, C. Kreibich, and V. Paxson. Measuring pay-per-install: the commoditization of malware distribution. In *Usenix security symposium*, pages 13–13, 2011.

[48] J. Caballero, P. Poosankam, C. Kreibich, and D. Song. Dispatcher: Enabling active botnet infiltration using automatic protocol reverse-engineering. In *Proceedings of the 16th ACM conference on Computer and communications security*, pages 621–634. ACM, 2009.

[49] D. S. Callaway, M. E. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Phy. rev. letters*, 85(25), 2000.

[50] Z. Carpou. Robots, pirates, and the rise of the automated takedown regime: Using the dmca to fight piracy and protect end-users. *Colum. JL & Arts*, 39:551, 2015.

[51] M. Castells. *The Internet galaxy: Reflections on the Internet, business, and society*. Oxford University Press on Demand, 2002.

[52] A. Celestini, G. Me, and M. Mignone. Tor marketplaces exploratory data analysis: The drugs case. In *International Conference on Global Security, Safety, and Sustainability*, pages 218–229. Springer, 2017.

[53] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali. Mean birds: Detecting aggression and bullying on twitter.

In *Proceedings of the 2017 ACM on web science conference*, pages 13–22, 2017.

[54] P. Chen, C. Huygens, L. Desmet, and W. Joosen. Advanced or not? a comparative study of the use of anti-debugging and anti-vm techniques in generic and targeted malware. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 323–336. Springer, 2016.

[55] C. Y. Cho, J. Caballero, C. Grier, V. Paxson, and D. Song. Insights from the inside: A view of botnet management from infiltration. *LEET*, 10:1–1, 2010.

[56] K.-K. R. Choo. The cyber threat landscape: Challenges and future research directions. *Computers & Security*, 30(8):719–731, Nov. 2011.

[57] N. Christin. Traveling the silk road: A measurement analysis of a large anonymous online marketplace. In *Proceedings of the 22nd international conference on World Wide Web*, pages 213–224, 2013.

[58] M. Christodorescu, S. Jha, S. A. Seshia, D. Song, and R. E. Bryant. Semantics-aware malware detection. In *IEEE Symposium on Security and Privacy*, 2005.

[59] Y. T. Chua, S. Parkin, M. Edwards, D. Oliveira, S. Schiffner, G. Tyson, and A. Hutchings. Identifying unintended harms of cybersecurity countermeasures. In *2019 APWG Symposium on Electronic Crime Research (eCrime)*, pages 1–15. IEEE, 2019.

[60] R. V. Clarke. Situational crime prevention: Theory and practice. *Brit. J. Criminology*, 20:136, 1980.

[61] R. V. Clarke, editor. *Situational crime prevention: successful case studies*. Harrow and Heston, Guilderland, NY, 2. ed edition, 1997. OCLC: 36877499.

[62] R. V. Clarke and D. B. Cornish. Modeling offenders' decisions: A framework for research and policy. *Crime and justice*, 6:147–185, 1985.

[63] R. V. G. Clarke. *Situational crime prevention*. Criminal Justice Press Monsey, NY, 1997.

[64] R. V. G. Clarke and G. R. Newman. *Outsmarting the terrorists*. Greenwood Publishing Group, 2006.

[65] R. V. G. Clarke and B. Webb. *Hot products: Understanding, anticipating and reducing demand for stolen goods*, volume 112. Citeseer, 1999.

[66] R. Clayton. How much did shutting down mccolo help. *Proc. of 6th CEAS*, 2009.

[67] E. P. Cockbain and G. Laycock. Crime science. Oxford University Press, 2017.

[68] L. E. Cohen and M. Felson. Social change and crime rate trends: A routine activity approach. *American sociological review*, 1979.

[69] R. Cohen and J. S. Hiller. Towards a Theory of Cyberplace: A Proposal for a New Legal Framework. *Richmond Journal of Law & Technology*, 10:41, 2003.

[70] E. Cooke, F. Jahanian, and D. McPherson. The zombie roundup: Understanding, detecting, and disrupting botnets. *SRUTI*, 5:6–6, 2005.

[71] D. B. Cornish. The procedural analysis of offending and its relevance for situational prevention. *Crime prevention studies*, 3:151–196, 1994.

[72] D. B. Cornish and R. V. Clarke. Understanding crime displacement: An application of rational choice theory. *Criminology*, 25(4):933–948, 1987.

[73] D. B. Cornish and R. V. Clarke. Opportunities, precipitators and criminal decisions: A reply to wortley's critique of situational crime prevention. *Crime prevention studies*, 16:41–96, 2003.

[74] T. Cresswell. *Place: an introduction*. John Wiley & Sons, 2014.

[75] G. De Maio, A. Kapravelos, Y. Shoshitaishvili, C. Kruegel, and G. Vigna. Pexy: The other side of exploit kits. In *Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, 2014.

[76] S. De Paoli. The engineer–criminologist and "the novelty of cybercrime": a situated genealogical study of timesharing systems. *Internet Histories*, 2(1-2):20–37, 2018.

[77] K. Dedel. *Sexual assault of women by strangers*. US Department of Justice, Office of Community Oriented Policing Services, 2011.

[78] L. Devriendt, B. Derudder, and F. Witlox. Cyberplace and Cyberspace: Two Approaches to Analyzing Digital Intercity Linkages. *Journal of Urban Technology*, 15(2):5–32, Aug. 2008.

[79] D. Dittrich. So you want to take over a botnet... In *Presented as part of the 5th {USENIX} Workshop on Large-Scale Exploits and Emergent Threats*, 2012.

[80] D. S. Dolliver. Evaluating drug trafficking on the tor network: Silk road 2, the sequel. *International Journal of Drug Policy*, 26(11):1113–1123, 2015.

[81] J. E. Eck and W. Spelman. Problem-solving: Problem-oriented policing in newport news. 1987.

[82] J. E. Eck and D. Weisburd. *Crime and place*, volume 4. Criminal Justice Press Monsey, NY, 1995.

[83] B. Edwards, S. Hofmeyr, S. Forrest, and M. Van Eeten. Analyzing and modeling longitudinal security data: Promise and pitfalls. In *Proceedings of the 31st Annual Computer Security Applications Conference*, pages 391–400, 2015.

[84] M. Edwards, G. N. S. de Tangil Rotaeche, C. Peersman, G. Stringhini, A. Rashid, and M. Whitty. The geography of online dating fraud. In *Workshop on Technology and Consumer Protection (ConPro)*, 2018.

[85] P. Ekblom. Gearing up against crime: A dynamic framework to help designers keep up with the adaptive criminal in a changing world. *International Journal of Risk Security and Crime Prevention*, 2:249–266, 1997.

[86] P. Ekblom. *Crime prevention, security and community safety using the 5Is framework*. Springer, 2010.

[87] J. Endrass, F. Urbaniok, L. C. Hammermeister, C. Benz, T. Elbert, A. Laubacher, and A. Rossegger. The consumption of internet child pornography and violent and sex offending. *BmC Psychiatry*, 9(1):43, 2009.

[88] B. Eshete, A. Alhuzali, M. Monshizadeh, P. A. Porras, V. N. Venkatakrishnan, and V. Yegneswaran. Ekhunter: A counter-offensive toolkit for exploit kit infiltration. In *Network and Distributed Systems Security Symposium (NDSS)*, 2015.

[89] G. Farrell and K. Pease. *Once bitten, twice bitten: Repeat victimisation and its implications for crime prevention*, volume 46. Home Office Police Research Group London, 1993.

[90] M. Finifter, D. Akhawe, and D. Wagner. An empirical study of vulnerability rewards programs. In *USENIX Security Symposium*, 2013.

[91] S. Ford, M. Cova, C. Kruegel, and G. Vigna. Analyzing and Detecting Malicious Flash Advertisements. In *2009 Annual Computer Security Applications Conference*, pages 363–372, Dec. 2009.

[92] A. M. Founta, D. Chatzakou, N. Kourtellis, J. Blackburn, A. Vakali, and I. Leontiadis. A unified deep learning architecture for abuse detection. In *Proceedings of the 10th ACM conference on web science*, pages 105–114, 2019.

[93] T. Gabor. The crime displacement hypothesis: An empirical examination. *Crime & Delinquency*, 27(3):390–404, 1981.

[94] R. Garside. *Crime, persistent offenders and the justice gap*. Crime and Society Foundation London, 2004.

[95] J. Glyde. Localities of crime in suffolk. *Journal of the Statistical Society of London*, 19(2):102–106, 1856.

[96] M. D. Goodman and S. W. Brenner. The emerging consensus on criminal conduct in cyberspace. *International Journal of Law and Information Technology*, 10(2):139–223, 2002.

[97] P. N. Grabosky. Virtual Criminality: Old Wine in New Bottles? *Social & Legal Studies*, 10(2):243–249, June 2001.

[98] C. Grier, L. Ballard, J. Caballero, N. Chachra, C. J. Dietrich, K. Levchenko, P. Mavrommatis, D. McCoy, A. Nappa, A. Pitsillidis, et al. Manufacturing compromise: the emergence of exploit-as-a-service. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 821–832. ACM, 2012.

[99] A.-M. Guerry. Essay on the moral statistics of france. *Paris: l'Academie des Sciences*, 1833.

[100] H. Haelterman. Situational crime prevention and supply chain security: theory for best practice. *CRISP Reports: Connecting Research in Security to Practice*, 2013.

[101] M. A. Harlev, H. Sun Yin, K. C. Langenheldt, R. Mukkamala, and R. Vatrapu. Breaking bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning. In *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[102] N. Henry and A. Powell. Technology-Facilitated Sexual Violence: A Literature Review of Empirical Research. *Trauma, Violence, & Abuse*, 19(2):195–208, Apr. 2018.

[103] C. Herley. Why Do Nigerian Scammers Say They are From Nigeria? *WEIS*, June 2012.

[104] C. Herley and D. Florêncio. Nobody Sells Gold for the Price of Silver: Dishonesty, Uncertainty and the Underground Economy. In T. Moore, D. Pym, and C. Ioannidis, editors, *Economics of Information Security and Privacy*, pages 33–53. Springer US, Boston, MA, 2010.

[105] R. Hesseling. Displacement: A review of the empirical literature. *Crime prevention studies*, 3(1):97–230, 1994.

[106] H. J. Highland. The brain virus: fact and fantasy. *Computers & Security*, 7(4):367–370, 1988.

[107] G. Hine, J. Onaolapo, E. De Cristofaro, N. Kourtellis, I. Leontiadis, R. Samaras, G. Stringhini, and J. Blackburn. Kek, cucks, and god emperor trump: A measurement study of 4chan's politically incorrect forum and its effects on the web. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017.

[108] T. J. Holt and A. M. Bossler. Examining the Applicability of Lifestyle-Routine Activities Theory for Cybercrime Victimization. *Deviant Behavior*, 30(1):1–25, Nov. 2008.

[109] T. Holz, C. Gorecki, K. Rieck, and F. C. Freiling. Measuring and Detecting Fast-Flux Service Networks. In *Network and Distributed Systems Security Symposium (NDSS)*, 2008.

[110] M. Horton-Eddison and M. Di Cristofaro. Hard interventions and innovation in crypto-drug markets: the escrow example. *Policy Brief*, 11, 2017.

[111] X. Hu, M. Knysz, and K. G. Shin. Measurement and analysis of global ip-usage patterns of fast-flux botnets. In *IEEE Conference on Computer Communications (INFOCOM)*, 2011.

[112] J. Huang, G. Stringhini, and P. Yong. Quit playing games with my heart: Understanding online dating scams. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 216–236. Springer, 2015.

[113] D. Hunter. Cyberspace as Place and the Tragedy of the Digital Anticommons. *California Law Review*, 91:439–520, 2003.

[114] E. M. Hutchins, M. J. Cloppert, and R. M. Amin. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research*, 1(1):80, 2011.

[115] C. C. Ife, T. Davies, S. J. Murdoch, and G. Stringhini. Bridging information security and environmental criminology research to better mitigate cybercrime. *arXiv preprint arXiv:1910.06380*, 2019.

[116] C. C. Ife, Y. Shen, S. J. Murdoch, and G. Stringhini. 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, 2019.

[117] A. Ioannou, J. Blackburn, G. Stringhini, E. D. Cristofaro, N. Kourtellis, and M. Sirivianos. From risk factors to detection and intervention: a practical proposal for future work on cyberbullying. *Behaviour & Information Technology*, Volume 37(Issue 3):Pages 258–266, Feb. 2018.

[118] T. Islam, I. Becker, R. Posner, P. Ekblom, M. McGuire, H. Borrion, and S. Li. A socio-technical and co-evolutionary framework for reducing human-related risks in cyber security and cybercrime ecosystems. *Communications in Computer and Information Science*, 2019.

[119] N. Jagpal, E. Dingle, J.-P. Gravel, P. Mavrommatis, N. Provos, M. A. Rajab, and K. Thomas. Trends and lessons from three years fighting malicious extensions. In *USENIX Security Symposium*, 2015.

[120] C. R. Jeffery. Crime prevention through environmental design. *American Behavioral Scientist*, 14(4):598–598, 1971.

[121] C. R. Jeffery. *Crime prevention through environmental design*, volume 524. Sage Publications California, 1977.

[122] S. D. Johnson, R. T. Guerette, and K. Bowers. Crime displacement: what we know, what we don't know, and what it means for crime reduction. *Journal of Experimental Criminology*, 10(4):549–571, 2014.

[123] J. Kamps and B. Kleinberg. To the moon: defining and detecting cryptocurrency pump-and-dumps. *Crime Science*, 7(1):18, 2018.

[124] A. Kapravelos, C. Grier, N. Chachra, C. Kruegel, G. Vigna, and V. Paxson. Hulk: Eliciting malicious behavior in browser extensions. In *USENIX Security Symposium*, 2014.

[125] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda. Cutting the gordian knot: A look under the hood of ransomware attacks. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 3–24. Springer, 2015.

[126] I. Koniaris, G. Papadimitriou, P. Nicopolitidis, and M. Obaidat. Honeypots deployment for the analysis and visualization of malware activity and malicious connections. In *2014 IEEE International Conference on Communications (ICC)*, pages 1819–1824, June 2014.

[127] P. Kotzias, L. Bilge, and J. Caballero. Measuring PUP prevalence and PUP distribution through pay-per-install services. In *USENIX Security Symposium*, pages 739–756, 2016.

[128] P. Kotzias and J. Caballero. An analysis of pay-per-install economics using entity graphs. In *Proceedings (online) of the Workshop on Economics and Information Security (WEIS)*, 2017.

[129] P. Kotzias, S. Matic, R. Rivera, and J. Caballero. Certified PUP: Abuse in authenticode code signing. In *ACM Conference on Computer and Communications Security (CCS)*, 2015.

[130] B. J. Kwon, J. Mondal, J. Jang, L. Bilge, and T. Dumitras. The dropper effect: Insights into malware distribution with downloader graph analytics. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1118–1129. ACM, 2015.

[131] B. J. Kwon, V. Srinivas, A. Deshpande, and T. Dumitraş. Catching worms, trojan horses and pups: Unsupervised detection of silent delivery campaigns. *arXiv preprint arXiv:1611.02787*, 2016.

[132] M. A. Lemley. Place and Cyberspace. *California Law Review*, 91:521–542, 2003.

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

[134] E. R. Leukfeldt. Comparing victims of phishing and malware attacks: Unraveling risk factors and possibilities for situational crime prevention. *arXiv preprint arXiv:1506.00769*, 2015.

[135] E. R. Leukfeldt and M. Yar. Applying Routine Activity Theory to Cybercrime: A Theoretical and Empirical Analysis. *Deviant Behavior*, 37(3):263–280, Mar. 2016.

[136] K. Levchenko, A. Pitsillidis, N. Chachra, B. Enright, M. Félegyházi, C. Grier, T. Halvorson, C. Kanich, C. Kreibich, H. Liu, et al. Click trajectories: End-to-end analysis of the spam value chain. In *IEEE Symposium on Security and Privacy*, 2011.

[137] C. Lever, P. Kotzias, D. Balzarotti, J. Caballero, and M. Antonakakis. A lustrum of malware network communication: Evolution and insights. In *IEEE Symposium on Security and Privacy*, 2017.

[138] B. Li, K. Roundy, C. Gates, and Y. Vorobeychik. Large-scale identification of malicious singleton files. In *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy*, pages 227–238, 2017.

[139] F. M. Llinares and S. D. Johnson. Cybercrime and place. In *The Oxford Handbook of Environmental Criminology*, chapter 38. 2018.

[140] J. Lynch. Identity Theft in Cyberspace: Crime Control Methods and Their Effectiveness in Combating Phishing Attacks II: Cyberlaw: A: Notes. *Berkeley Technology Law Journal*, 20:259–300, 2005.

[141] E. Mariconti, J. Onaolapo, S. S. Ahmad, N. Nikiforou, M. Egele, N. Nikiforakis, and G. Stringhini. What's in a name? understanding profile name reuse on twitter. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1161–1170, 2017.

[142] E. Mariconti, G. Suarez-Tangil, J. Blackburn, E. De Cristofaro, N. Kourtellis, I. Leontiadis, J. L. Serrano, and G. Stringhini. "You Know What to Do": Proactive Detection of YouTube Videos Targeted by Coordinated Hate Attacks. *arXiv:1805.08168 [cs]*, May 2018. arXiv: 1805.08168.

[143] D. Massey. *Space, place and gender*. John Wiley & Sons, 2013.

[144] G. Mba, J. Onaolapo, G. Stringhini, and L. Cavallaro. Flipping 419 cybercrime scams: Targeting the weak and the vulnerable. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1301–1310, 2017.

[145] M. McGuire and S. Dowling. Cyber crime: A review of the evidence. *Summary of key findings and implications. Home Office Research report*, 75, 2013.

[146] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage. A fistful of bitcoins: characterizing payments among

men with no names. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 127–140. ACM, 2013.

[147] D. Moore, C. Shannon, and K. Claffy. Code-red: a case study on the spread and victims of an internet worm. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 273–284, 2002.

[148] J. A. Morales, A. Al-Bataineh, S. Xu, and R. Sandhu. Analyzing and exploiting network behaviors of malware. In *International conference on security and privacy in communication systems*, pages 20–34. Springer, 2010.

[149] S. J. Murdoch and G. Danezis. Low-cost traffic analysis of tor. In *2005 IEEE Symposium on Security and Privacy (S&P'05)*, pages 183–195. IEEE, 2005.

[150] Y. Nadji, M. Antonakakis, R. Perdisci, D. Dagon, and W. Lee. Beheading hydras: performing effective botnet takedowns. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 121–132. ACM, 2013.

[151] Y. Nadji, R. Perdisci, and M. Antonakakis. Still beheading hydras: Botnet takedowns then and now. *IEEE Transactions on Dependable and Secure Computing*, 14(5):535–549, 2015.

[152] A. Nappa, M. Z. Rafique, and J. Caballero. Driving in the Cloud: An Analysis of Drive-by Download Operations and Abuse Reporting. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, Lecture Notes in Computer Science, pages 1–20. Springer, Berlin, Heidelberg, July 2013.

[153] J. N. Navarro and J. L. Jasinski. Going Cyber: Using Routine Activities Theory to Predict Cyberbullying Experiences. *Sociological Spectrum*, 32(1):81–94, Jan. 2012.

[154] T. Nelms, R. Perdisci, M. Antonakakis, and M. Ahamad. Webwitness: Investigating, categorizing, and mitigating malware download paths. In *USENIX Security Symposium*, 2015.

[155] T. Nelms, R. Perdisci, M. Antonakakis, and M. Ahamad. Towards measuring and mitigating social engineering software download attacks. In *USENIX Security Symposium*, pages 773–789, 2016.

[156] O. Newman. Crime prevention through urban design defensible space. *The Mcmillan Company, New York*, 1972.

[157] Y. Y. Ng, H. Zhou, Z. Ji, H. Luo, and Y. Dong. Which android app store can be trusted in china? In *Computer Software and Applications Conference (COMPSAC), 2014 IEEE 38th Annual*, pages 509–518. IEEE, 2014.

[158] J. Onaolapo, E. Mariconti, and G. Stringhini. What happens after you are pwnd: Understanding the use of leaked webmail credentials in the wild. In *Proceedings of the 2016 Internet Measurement Conference*, pages 65–79, 2016.

[159] S. Parkin and Y. T. Chua. Refining the blunt instruments of cybersecurity: A framework to coordinate prevention and preservation of behaviours. In *International Workshop on Socio-Technical Aspects in Security and Trust*, pages 23–42. Springer, 2020.

[160] S. Pastrana and G. Suarez-Tangil. A first look at the crypto-mining malware ecosystem: A decade of unrestricted wealth. In *Proceedings of the Internet Measurement Conference*, pages 73–86, 2019.

[161] K. Pease et al. *Repeat victimisation: Taking stock*, volume 90. Home Office Police Research Group London, 1998.

[162] P. Peng, L. Yang, L. Song, and G. Wang. Opening the blackbox of virustotal: Analyzing online phishing scan engines. In *Proceedings of the Internet Measurement Conference*, pages 478–485. ACM, 2019.

[163] D. Plohmann, K. Yakdan, M. Klatt, J. Bader, and E. Gerhards-Padilla. A comprehensive measurement study of domain generating malware. In *USENIX Security Symposium*, 2016.

[164] B. Poyner. *Design against crime: Beyond defensible space*. Butterworths London, 1983.

[165] N. Provos, D. McNamee, P. Mavrommatis, K. Wang, N. Modadugu, et al. The ghost in the browser: Analysis of web-based malware. In *HotBots*, 2007.

[166] A. Quetelet. *A Treatise on Man and the Development of His Faculties: Now First Translated Into English*. William and Robert Chambers, 1842.

[167] T. A. Reppetto. Crime prevention and the displacement phenomenon. *Crime & Delinquency*, 22(2):166–177, 1976.

[168] B. W. Reyns. A situational crime prevention approach to cyberstalking victimization: Preventive tactics for Internet users and online place managers. *Crime Prevention and Community Safety*, 12(2):99–118, Apr. 2010.

[169] B. W. Reyns. Online routines and identity theft victimization: Further expanding routine activity theory beyond direct-contact offenses. *Journal of Research in Crime and Delinquency*, 50(2):216–238, 2013.

[170] L. D. Roberts, D. Indermaur, and C. Spiranovic. Fear of Cyber-Identity Theft and Related Fraudulent Activity. *Psychiatry, Psychology and Law*, 20(3):315–328, June 2013.

[171] D. K. Rossmo. *Geographic profiling*. CRC press, 1999.

[172] C. Rossow, C. Dietrich, and H. Bos. Large-scale analysis of malware downloaders. In *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*. 2013.

[173] T. Saito. A microeconomic analysis of bitcoin and illegal activities. In *Handbook of Digital Currency*, pages 231–248. Elsevier, 2015.

[174] R. Sampson, J. E. Eck, and J. Dunham. Super controllers and crime prevention: A routine activity explanation of crime prevention success and failure. *Security Journal*, 23(1):37–51, 2010.

[175] S. Schiavoni, F. Maggi, L. Cavallaro, and S. Zanero. Phoenix: Dga-based botnet tracking and intelligence. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 192–211. Springer, 2014.

[176] B. Schneier. Attack trees. *Dr. Dobb's journal*, 24(12):21–29, 1999.

[177] M. Sebastián, R. Rivera, P. Kotzias, and J. Caballero. Avclass: A tool for massive malware labeling. In *International Symposium on Research in Attacks, Intrusions, and Defenses (RAID)*, 2016.

[178] J. Serra, I. Leontiadis, D. Spathis, G. Stringhini, J. Blackburn, and A. Vakali. Class-based prediction errors to detect hate speech with out-of-vocabulary words. In *Proceedings of the first workshop on abusive language online*, pages 36–40, 2017.

[179] C. R. Shaw and H. D. McKay. *Juvenile delinquency and urban areas.* 1942.

[180] L. W. Sherman. Defiance, deterrence, and irrelevance: A theory of the criminal sanction. *Journal of research in Crime and Delinquency*, 30(4):445–473, 1993.

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

[182] A. K. Sood and R. J. Enbody. Malvertising – exploiting web advertising. *Computer Fraud & Security*, 2011(4):11–16, Apr. 2011.

[183] A. K. Sood and R. J. Enbody. Crimeware-as-a-service—A survey of commoditized crimeware in the underground market. *International Journal of Critical Infrastructure Protection*, 6(1):28–38, Mar. 2013.

[184] K. Soska and N. Christin. Automatically detecting vulnerable websites before they turn malicious. In *USENIX Security Symposium*, pages 625–640, 2014.

[185] K. Soska and N. Christin. Measuring the longitudinal evolution of the online anonymous marketplace ecosystem. In *USENIX Security Symposium*, pages 33–48, 2015.

[186] B. Stone-Gross, M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel, and G. Vigna. Your botnet is my botnet: analysis of a botnet takeover. In *ACM conference on Computer and communications security (CCS)*, 2009.

[187] B. Stone-Gross, M. Cova, B. Gilbert, R. Kemmerer, C. Kruegel, and G. Vigna. Analysis of a botnet takeover. *IEEE Security & Privacy*, 9(1):64–72, 2011.

[188] B. Stone-Gross, T. Holz, G. Stringhini, and G. Vigna. The underground economy of spam: A botmaster's perspective of coordinating large-scale spam campaigns. *LEET*, 11:4–4, 2011.

[189] B. Stone-Gross and P. Khandhar. Dyre Banking Trojan Threat Analysis - intelligence, dell secureworks counter threat unit™ threat. *https://www.secureworks.com/research/dyre-banking-trojan*. [Accessed online: 11-September-2020].

[190] B. Stone-Gross, C. Kruegel, K. Almeroth, A. Moser, and E. Kirda. Fire: Finding rogue networks. In *Annual Computer Security Applications Conference (ACSAC)*, pages 231–240. IEEE, 2009.

[191] G. Stringhini. Adversarial behaviours knowledge area. In *Cyber Security Body of Knowledge*. 2019.

[192] G. Stringhini, O. Hohlfeld, C. Kruegel, and G. Vigna. 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)*, 2014.

[193] G. Stringhini, C. Kruegel, and G. Vigna. Shady paths: Leveraging surfing crowds to detect malicious web pages. In *ACM conference on Computer and communications security (CCS)*, 2013.

[194] G. Stringhini, Y. Shen, Y. Han, and X. Zhang. Marmite: Spreading malicious file reputation through download graphs. In *Proceedings of the 33rd Annual Computer Security Applications Conference*, pages 91–102. ACM, 2017.

[195] F. Sussan, S. Gould, and S. Weisfeld-Spolter. Location, location, location: the relative roles of virtual location, online word-of-mouth (ewom) and advertising in the new-product adoption process. *ACR North American Advances*, 2006.

[196] Symantec. Dyre: Operations of bank fraud group grind to halt following takedown. *https://www.symantec.com/connect/blogs/dyre-operations-bank-fraud-group-grind-halt-following-takedown*, 2016. [Accessed online: 11-August-2017].

[197] T. Taylor, X. Hu, T. Wang, J. Jang, M. P. Stoecklin, F. Monrose, and R. Sailer. Detecting malicious exploit kits using tree-based similarity searches. In *ACM Conference on Data and Application Security and Privacy*, 2016.

[198] R. Telang. Does online piracy make computers insecure? evidence from panel data. *Evidence from Panel Data (March 12, 2018)*, 2018.

[199] K. Thomas, E. Bursztein, C. Grier, G. Ho, N. Jagpal, A. Kapravelos, D. McCoy, A. Nappa, V. Paxson, P. Pearce, et al. Ad injection at scale: Assessing deceptive advertisement modifications. In *IEEE Symposium on Security and Privacy*, 2015.

[200] K. Thomas, J. A. E. Crespo, R. Rasti, J.-M. Picod, C. Phillips, M.-A. Decoste, C. Sharp, F. Tirelo, A. Tofigh, M.-A. Courteau, et al. Investigating commercial pay-per-install and the distribution of unwanted software. In *USENIX Security Symposium*, pages 721–739, 2016.

[201] K. Thomas, D. Huang, D. Wang, E. Bursztein, C. Grier, T. J. Holt, C. Kruegel, D. McCoy, S. Savage, and G. Vigna. Framing dependencies introduced by underground commoditization. 2015.

[202] R. S. Tokunaga. Following you home from school: A critical review and synthesis of research on cyberbullying victimization. *Computers in Human Behavior*, 26(3):277–287, May 2010.

[203] E. Tranos and P. Nijkamp. The Death of Distance Revisited: Cyber-Place, Physical and Relational Proximities. *Journal of Regional Science*, 53(5):855–873, Dec. 2013.

[204] J. Van Wilsem. 'bought it, but never got it'assessing risk factors for online consumer fraud victimization. *European Sociological Review*, 29(2):168–178, 2011.

[205] M. Vasek and T. Moore. Identifying Risk Factors for Webserver Compromise. In *Financial Cryptography and Data Security*, Lecture Notes in Computer Science, pages 326–345. Springer, Berlin, Heidelberg, Mar. 2014.

[206] M. Vasek, J. Wadleigh, and T. Moore. Hacking Is Not Random: A Case-Control Study of Webserver-Compromise Risk. *IEEE Transactions on Dependable and Secure Computing*, 13(2):206–219, Mar. 2016.

[207] D. Wall. *Cybercrime: The transformation of crime in the information age*, volume 4. Polity, 2007.

[208] Z. Wang, H. Li, Q. Li, W. Li, H. Zhu, and L. Sun. Towards ip geolocation with intermediate routers based on topology discovery. *Cybersecurity*, 2(1):1–14, 2019.

[209] B. Wellman. Computer Networks As Social Networks. *Science*, 293(5537):2031–2034, Sept. 2001.

[210] B. Wellman. Physical Place and Cyberplace: The Rise of Personalized Networking. *International Journal of Urban and Regional Research*, 25(2):227–252, June 2001.

[211] M. T. Whitty and T. Buchanan. The Online Romance Scam: A Serious Cybercrime. *Cyberpsychology, Behavior, and Social Networking*, 15(3):181–183, Feb. 2012.

[212] S. E. Wick, C. Nagoshi, R. Basham, C. Jordan, Y. K. Kim, A. P. Nguyen, and P. Lehmann. Patterns of cyber harassment and perpetration among college students in the united states: A test of routine activities theory. *International Journal of Cyber Criminology*, 11(1):24–38, 2017.

[213] J. Q. Wilson and G. L. Kelling. Broken windows. *Atlantic monthly*, 249(3):29–38, 1982.

[214] J. Wolak, D. Finkelhor, K. J. Mitchell, and M. L. Ybarra. Online "predators" and their victims: Myths, realities, and implications for prevention and treatment. 2010.

[215] R. Wortley. A Classification of Techniques for Controlling Situational Precipitators of Crime. *Security Journal*, 14(4):63–82, Oct. 2001.

[216] R. Wortley and N. Tilley. Theories for situational and environmental crime prevention. Springer, 2014.

[217] S. Yadav, A. K. K. Reddy, A. N. Reddy, and S. Ranjan. Detecting algorithmically generated malicious domain names. In *ACM Conference on Internet Measurement (IMC)*, 2010.

[218] W. Yan, Z. Zhang, and N. Ansari. Revealing packed malware. *ieee seCurity & PrivaCy*, 6(5):65–69, 2008.

[219] M. Yar. The Novelty of 'Cybercrime': An Assessment in Light of Routine Activity Theory. *European Journal of Criminology*, 2(4):407–427, Oct. 2005.

[220] E. Yiallourou, R. Demetriou, and A. Lanitis. On the detection of images containing child-pornographic material. In *2017 24th International Conference on Telecommunications (ICT)*, pages 1–5, Limassol, Cyprus, May 2017. IEEE.

[221] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1):22–32, 2014.

[222] A. Zarras, A. Kapravelos, G. Stringhini, T. Holz, C. Kruegel, and G. Vigna. The dark alleys of madison avenue: Understanding malicious advertisements. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 373–380. ACM, 2014.

[223] P. G. Zimbardo, C. Haney, W. C. Banks, and D. Jaffe. *Stanford prison experiment*. Zimbardo, Incorporated, 1971.