An empirical analysis of ZeuS C&C lifetime

Carlos Gañán
Delft University of Technology
The Netherlands
c.h.g.hernandezganan@tudelft.nl

Orcun Cetin
Delft University of Technology
The Netherlands
F.O.Cetin@tudelft.nl

Michel van Eeten
Delft University of Technology
The Netherlands
M.J.G.vanEeten@tudelft.nl

ABSTRACT

Botnets continue to pose a significant threat to network-based applications and communications over the Internet. A key mitigation strategy has been to take down command and control infrastructure of the botnets. The efficiency of those mitigation methods has not been extensively studied. In this paper we investigate several observable characteristics of botnet command and controls (C&C) and estimate the variability in the survival rate of these C&Cs and the factors that are related to such variability. Furthermore, we show that different type of mitigation efforts have different impact. Kaplan-Meier analysis is performed to evaluate C&C survival ratios in the particular case of the ZeuS botnet. Using a lasso penalized Cox regression model, we identify the factors that influence the lifetime of a C&C. Location, malware family type, registrar, hosting type and popularity are the fundamental factors that explain this variability. Our results show that location and type of hosting are the two factors that affect more significantly the C&C lifetime. Thus, ZeuS C&Cs in certain regions of Asia are prone to stay online longer that those located in Europe.

Keywords

ZeuS malware, C2C lifetime, survival analysis

Categories and Subject Descriptors

K.6.5 [**Management of computing and information systems**]: Security and Protection; K.4.2 [**Computers and society**]: Social Issues

1. INTRODUCTION

A botnet comprises a set of malware-infected machines connected to the Internet that communicate to accomplish distributed criminal activities. Botnets support a wide array of criminal business models which represent a severe threat to the Internet economy, individuals' privacy, and national security.

One of the main strategies for combating botnets has been to disrupt the ability of the criminals to commandeer the botnet's resources. The criminals control the infected machines either directly via built-in commands, or indirectly via a command and control (C&C) infrastructure. Roughly speaking, C&C can rely on a centralized or a peer-to-peer architecture, enhanced by complementary techniques to ensure redundancy and resiliency against take-down attempts by defenders.

It is common to view C&C takedown as a primarily technical problem of reverse engineering the botnet's communication mechanisms and then disrupting it, by shutting down or manipulating key resources. What this view obscures, however, is that technological advances alone has proven insufficient to eradicate botnets [38]. Attempts to mitigate the proliferation of bots also have to take into account the incentives of the market actors that are needed to takedown criminal resources. Incentives are not just monetary, but also include reputation effects, legal constraints and the political environment in which these actors operate [37]. Understanding and aligning the incentives is as important as improving the technology in addressing cybersecurity threats.

Any attempt to improve these mitigation efforts will require identification of factors influencing the lifetime of botnet C&C. These factors help us to understand possible causes and identify pro-active and re-active countermeasures. To check the impact of both technological and socio-economic efforts towards mitigating botnets, we assess the lifetime duration of C&Cs depending on their characteristics. In this paper we use the ZeuS botnet [11] as case study to analyze these characteristics.

Zeus, Citadel and other variants of Zeus malware dominate the financial malware threat landscape and are responsible for huge business and costumer financial losses. According to a security report published in 2009, Zeus Crimeware along infected an estimated 3.6 million computers in the United States [11]. In order to mitigate the threat of the family of Zeus botnets, several takedown actions have been executed by both FBI and Microsoft. Some of them had an impact, but none of them brought down the botnets fully successfully.

Using three different data sources, we gather URLs related to C&Cs and analyze their characteristics, i.e., type of web server and web technologies. We investigate the relation between these characteristics and the C&C lifetime. Our results show that C&Cs running content management systems are less likely to host ZeuS than those are not running one, and that servers running *Tengine* are more likely to stay

ASIA CCS'15, April 14–17, 2015, Singapore..
Copyright © 2015 ACM 978-1-4503-3245-3/15/04 ...$15.00.
http://dx.doi.org/10.1145/2714576.27145794.

compromise longer than those running common servers like *Apache* or *nginx*.

Furthermore, we analyze the effectiveness of performed mitigating operations. We investigate the impact of four major mitigation efforts towards reducing the number of active C&Cs. Traditional time-series analysis shows significant differences among the take-down operations depending on the parties involved. Taking down a fraud friendly autonomous system is shown to be the most effective measure.

Survival analysis using Kaplan-Meier [22] estimates and Cox proportional hazards model [14] are used to determine potential differences in C&Cs lifetime, caused by different factors. A penalized likelihood algorithm is used to identify the importance of each factor and quantify their influence on the C&C's lifetime. Results show that the C&C's location and the hosting type are the main factors affecting the survival rate.

The rest of the paper is organized as follows. Section 2 describes the data collection methodology. Section 3 presents technological characteristics of ZeuS C&Cs. Section 4 analyzes the impact of different mitigation efforts. In Section 5 we evaluate C&C survival rate depending on several socio-demographic factors. Section 6 presents a lasso penalized Cox regression model to explain the importance of each factor. In section 7 we discus the related work and in section 8 we conclude.

2. DATA ON ZEUS COMMAND AND CONTROL SERVERS

2.1 Data Collection Methodology

The data set consists of 48,104 URLs associated with real C&C servers observed in the wild during approximately three years (the observation period varies depending on the data source). We use three independent data sources to collect ZeuS command and control URLs:

- *ZeuS Tracker* [5]: is a C&C panel tracker that provides the ability to track ZeuS command and control services and hosts of ZeuS files. It aims to provide system administrators with the possibility to block well-known ZeuS hosts and avoid ZeuS infections in their networks. The data-set consists of 18,889 URLs associated with C&Cs that were online at some point between 2009Q1 and 2014Q3. Let $\mathcal{Z}$ be the set of C&Cs from ZeuS tracker.

- *Cybercrime tracker* [1]: is a C&C panel tracker that lists the administration interfaces of certain in-the-wild botnets including ZeuS. Since 2013, this tracker is also integrated in Virus Total [3]. The data-set consists of 1,890 URLs associated with C&Cs that were online at some point between 2011Q4 and 2014Q3. Let $\mathcal{C}$ be the set of C&Cs from Cybercrime tracker.

- *Private honeypots*: consists of a list of botnet C&C servers captured by a company that specializes in threats intelligence using honeypots located all over the world. The data-set consists of 144,625 captured ZeuS configuration files. The configuration files were collected over a period of around two years (2011Q4-2013Q1) using two different methods: they are gathered by running live ZeuS samples or by emulating the malware to download configuration files. Let $\mathcal{P}$ be the set of C&Cs captured by these honeypots.

Table 1 shows the number of C&Cs per data source and the overlap among them. Most of the C&Cs in our data-set were captured by the private honeypots (56.80%) while Cybercrime tracker only captured a small amount (3.92%). It is also worth noting that there exists some overlap between the different data-sets. This overlap is insignificant between Cybercrime tracker and the private honeypots, while around 16.07% percent of the C&Cs present in the private data-set are also present in ZeuS tracker. On the other hand, 38,866 of the captured C&Cs are only present in one of the sources.

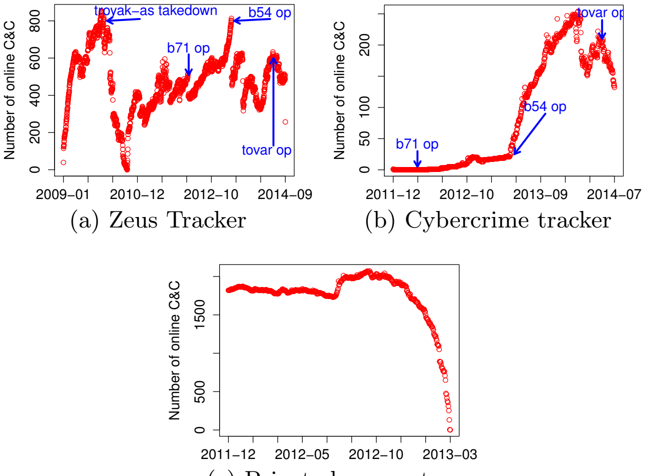
	Cardinality
$\mathcal{Z}$	18,889
$\mathcal{P}$	27,326
$\mathcal{C}$	1,890
$\mathcal{Z} \cap \mathcal{P}$	4,394
$\mathcal{Z} \cap \mathcal{C}$	119
$\mathcal{C} \cap \mathcal{P}$	8
$\mathcal{Z} \cap \mathcal{P} \cap \mathcal{C}$	6

Table 1: Number of C&Cs per source

From each C&C, we identify six different features: lifetime, ZeuS family type, location, hosting provider/AS number, registrar and popularity.

2.2 Summary statistics

We first discuss how the number of online C&Cs has evolved over time (see Fig. 1). The average number of online C&Cs (μ) varies highly depending on the data-set: Zeus tracker $\mu = 486.70$ ($\sigma = 164.15$), Cybercrime tracker $\mu = 84.77$ ($\sigma = 89.28$) and the private honeypots $\mu = 1,756.00$ ($\sigma = 328.01$). Though we observe that the different data sources captured different number of C&Cs, they all show a similar trend: the number of online C&Cs is converging to a common value independently of the data source. As can be seen in Fig. 1, a clear decreasing trend appeared in last quarter of 2014.



(a) Zeus Tracker

(b) Cybercrime tracker

(c) Private honeypots

Figure 1: Online C2C time series per source

We then study where these C&Cs are located (see Fig. 2). Our global data-set covers C&Cs located in 135 different countries. The top 10 countries in terms of hosting more C&Cs are United Stated (36.26%), Germany (14.05%), Russia (10.30%), Canada (4.25%), China (3.21%), Netherlands (3.11%), Ukraine (3.02%), United Kingdom (2.63%), India (2.35%) and France (1.45%). These 10 countries account for more than 80% of the total number of C&Cs, while the remaining 125 countries only account for less than 20%. This evidences that the number of C&Cs is related to the hosting industry market, i.e., a larger hosting industry increases the probability of hosting a C&C.

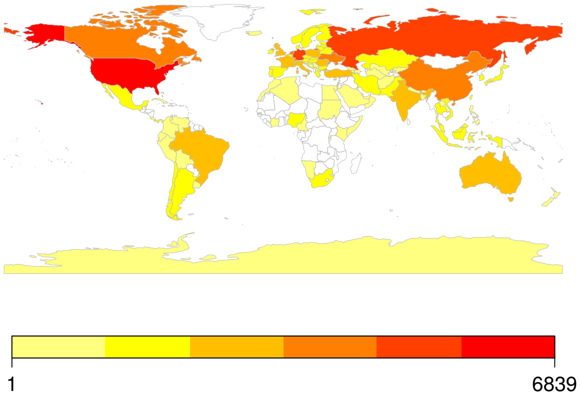


Figure 2: Number of C&C per country

Finally, we also analyze the distribution of the C&Cs among the different autonomous systems (AS). Table 2 shows the number of unique domains per data source. By comparing Table 1 and Table 2, we determine that several C&Cs are hosted in the same domain at some point in time. Moreover, it is also surprising that the number of unique IPs is smaller than the number of unique domains. That suggests that different domains share the same IP which evidences that shared hosting servers are being exploited as C&Cs. On the other hand, by comparing the number of ASes, we notice what part of the cyberspace is covered by each data source. ZeuS tracker is the most comprehensive in that sense, as it covers many more ASes than the other sources.

	# domains	Unique IPs	# ASes
Zeus Tracker	18,093	5,244	1181
Cybercrime tracker	1,647	1,135	449
Private honeypots	12,952	3,047	821

Table 2: Characteristics of the C&C domains per source

3. C&C CHARACTERISTICS

To further analyze why those particular servers were hosting C&C, we investigate their technical features. To that end, we scan all the C&Cs URLs using Netcraft toolbar [2].

First we analyze the uptime depending on the type of server. We define *uptime* as the amount of time since the C&C is detected until the C&C is taken down. Table 3 shows that there are appreciable differences in the uptime among different types of server. Though the frequency of the servers is related with their usage, the uptime is not. Tengine is the 9th top technology among web servers [4], but it hosts C&Cs for three times longer than traditional web servers such as Apache or Microsoft-IIS. We can also observe some significant differences among the different type of servers that host ZeuS C&Cs and the market share of these type of servers. That might indicate that these type of servers are more vulnerable or bot herders present some kind a preference towards infecting these type of web servers.

Server type	Avg. Uptime(h)	Observed	Expected)
Apache	1816.50	51.74%	59.4%
nginx	3276.78	27.92%	22.30%
Microsoft-IIS	2115.24	11.56%	13.5%
Tengine	6057.76	2.32%	0.17%
LiteSpeed	1142.90	1.66%	0.88%
cloudflare-nginx	2719.66	0.56%	0.58%
uServ	3191.97	0.25%	0.42%
lighttpd	2003.52	0.22%	0.75%
Microsoft-HTTPAPI	2993.71	0.17%	1.39%
YTS	1676.29	0.16%	0.04%
Squeegit	411.14	0.15%	0.02%
squid	1739.29	0.15%	0.20%
Apache-Coyote	1513.44	0.10%	0.29%
IdeaWebServer	1136.69	0.10%	0.40%
ghs	3913.35	0.08%	0.04%
IIS	3807.70	0.05%	1.07%
BladeStorm	2239.00	0.04%	0.04%
Zeus	666.00	0.04%	0.11%
PCX	950.20	0.03%	0.02%
Sun-ONE-Web-Server	34.40	0.03%	0.03%

Table 3: Uptime versus server type

Next we analyze the uptime depending on the type of content management system (CMS). Firstly, it is worth mentioning that only 36.77% of the C&Cs were hosted in a server running a CMS. The existence of sinkholes may have led to underestimate the number of C&Cs running on top of a CMS. However, the comparison among the different CMS types is representative of the ecosystem. Table 4 shows that there are important differences in the uptime among different types of CMS. Again we notice a similar trend that with the server type, i.e., some of the less popular CMSes host C&Cs for longer periods of time than the most popular ones. For instance, servers using Contao CMS lived ten times longer in average than those using WordPress.

CMS	Avg. Uptime(h)	Observed	Expected
WordPress	1164.96	54.65%	61.1%
Joomla	1293.88	31.42%	7.9%
Drupal	1421.91	2.16%	5.1%
Magento eCommerce	921.37	1.40%	2.7 %
Bitrix Site Manager	1077.08	1.08%	1.0%
osCommerce	690.87	1.02%	0.6%
zencart	4027.80	0.89%	0.4%
vBulletin	699.58	0.70%	1.1%
TYPO3	1015.38	0.57%	1.6%
Expression Engine	1256.30	0.51%	0.6%
OpenCart Shopping Cart	1214.93	0.51%	0.9%
phpBB	5564.10	0.38%	0.6%
Datalife Engine	4977.00	0.25%	0.9%
Google Sites	6964.00	0.25%	0.1%
DotNetNuke	2228.33	0.19%	0.9%
ECShop Shopping Cart	891.33	0.19%	0.2%
Simple Machines Forum	3070.00	0.19%	0.3%
concrete5	157.00	0.13%	0.2%
Contao CMS	11830.50	0.13%	0.2%

Table 4: Uptime versus content management system

Finally, we also analyze the different technologies searching for some pattern that would indicate that something is out of the ordinary. Table 5 describes the main statistics of the different technologies used by the C&Cs. Results show that 92.73% of the servers hosting C&C had some javascript code. This might indicate either that this technology represents a vulnerability or that certain ZeuS variants need javascript to run their C&C. In addition, sites that allow secure communication are less attractive to host a C&C and, in addition, they are cleaned faster.

	PHP	Javascript	Flash	SSL	Ajax	Perl
Observed Percentage	75.39%	92.73%	4.02%	19.08%	4.68%	14.04%
Expected Percentage	82.1%	88.1%	12.7%	48.9%	<0.1%	0.6%
Avg. Uptime(h) WITH	1993.36	2073.78	1259.27	1259.39	1381	1496.67
Avg. Uptime(h) WITHOUT	1898.04	1317.49	2050.66	2137.40	2031	2047.21

Table 5: Descriptive statistics C&C technologies

4. IMPACT OF MITIGATION MEASURES

In Section 2, we showed the evolution of the number of online C&Cs. In this section we analyze the impact of the different operations that took place to reduce that number.

During the period under observation, 4 interventions took place (see Fig. 1): troyak-AS takedown, b54 operation, b71 operation and tovar operation. As can be seen in Fig. 1a, on March 2010 the number of active ZeuS C&C servers dropped drastically. This drop was due to the takedown of Troyak-AS (AS50215), the upstream provider for the six worst ZeuS hosting ISPs. On March 2012, in operation b71, Microsoft obtained an ex parte temporary restraining order to seize domain names from botnet operators (in particular Zeus, SpyEye and Ice-IX variants). Again this operation caused a drop in the number of online C&Cs as shown in Fig. 1a. Note that the impact of this operation was not captured by cybercrime tracker (see Fig. 1b). This could be due to the fact that Microsoft took C&Cs that were operated mainly by researchers and security firms. Similarly, on June 2013, in operation b54, Microsoft in a collaborative action with the FBI took down several Citadel C&Cs that resulted in a significant disruption. The last operation -operation tovar- was performed on June 2014 by global law enforcement in conjunction with private partners to dismantle the Gameover Zeus. Note that these interventions only had a temporary impact on the botnet infrastructure but in any case it was possible to fully take down the botnet.

To assess the impact of the different interventions we model Zeus Tracker time series following the framework introduce by Box and Tiao [12]. This framework allows to measure how the interventions affected the number of online C&Cs by changing the mean function or the trend of the time series. To that end, we propose a basic univariate time series model is the Gaussian autoregressive moving average model:

$$\Phi(B)Z_t = \theta_0 + \theta(B)a_t, \tag{1}$$

where $\Phi(B) = 1 - \Phi_1 B - \ldots - \Phi_p B^p$ and $\theta(B) = 1 - \theta_1 B - \ldots - \theta_q B^q$ are polynomials in B of degrees p and q, respectively, θ_0 is a constant, B is the backshift operator such that $BZ_t = Z_{t-1}$, and a_t is a sequence of independent Gaussian variates with mean zero and variance σ_a^2.

More generally, we model the change in the mean function by an ARMAX-type specification so that it can handle intervention analysis and outliers. We consider two types of intervention: Level Shift and Transient Change. It is assumed that the intervention affects the mean function of the process, with the deviation from the unperturbed mean function modeled as the sum of the outputs of an ARMA filter of a number of covariates; the deviation is known as the transfer function. We consider two transient changes corresponding to the take-down Troyak-AS and operation tovar; and two level shifts corresponding to operations b51 and b54. The level shifts are included in the ARMA model from Eq. (1) by means of the indicator variable I_t that equals 1 in the intervention date and 0 otherwise. Transient changes are modeled as the sum of two ARMA filters of the corresponding indicator variable. Hence the transfer function equals:

$$\omega_0 I_t + \frac{\omega_1}{(1 - \omega_2 B)} I_t, \tag{2}$$

where I_t is a pulse function and ω_i is the deviation from the original time series caused by the intervention.

Table 6 shows the coefficients of the ARMAX model that best fit the time series series. To assess the goodness of the fit we compute the autocorrelation function of the residuals and the p-values. Fig. 3 shows that the standardized residuals do not show clusters of volatility. Similarly, the autocorrelation function shows no significant autocorrelation between the residuals and the p-values for the Ljung–Box statistics are all large, indicating that the residuals are patternless.

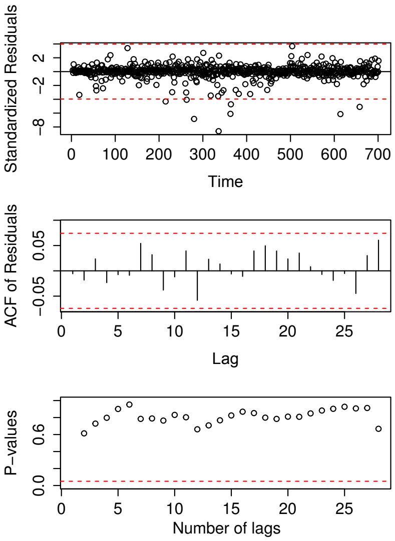


Figure 3: Diagnosis of the model fit

To compare the different interventions we analyze the value of the w_i coefficients. Comparing the ω_0 coefficients we can observe the change in the mean number of online C&Cs per intervention. The results corroborate the visual analysis, i.e., the take-down of Troyak-AS was the intervention that had a higher impact. The value of ω_2 determines how fast the effect of the intervention decays to zero. Again, the take-down of Troyak-AS had a longer persisting impact than the rest of the interventions.

5. FACTORS AFFECTING C&C UPTIME

In the previous sections, we have presented the technical characteristics of severs hosting C&Cs and showed how

		Troyak-AS			b71	b54	Tovar operation		
θ	Φ	ω_0	ω_1	ω_2	ω_0	ω_0	ω_0	ω_1	ω_2
-0.4031	0.5222	-181.3694	181.1214	0.0202	-17.0044	-19.8002	-61.7158	49.61	-0.2058
(0.1174)	(0.1073)	(44.5913)	(55.6024)	(0.0045)	(16.1217)	(17.3852)	(44.5918)	(23.1023)	(0.0125)

Table 6: Parameters (value and standard error) of the ARMAX model considering interventions

these affect the uptime. In this section, we analyze the impact of not only technical factors but also socio-demographic variables on the uptime of the C&Cs.

There exist several methods by which a botnet can be taken down [13]. Traditional methods include:

- Hosting provider de-peered
- Server hosting botnet cleans up/kicks off
- Public IRC servers takedown, expiration free web hosting
- Compromised host cleaned up/rebuilt
- DNS Revoked
- IP of C&C server blacklisted

These different take-down mechanisms involve different parties, e.g. registrars or hosting providers. In the following we evaluate and interpret the impact that have the different entities involved in these methods on the lifetime of the C&C. We perform a survival analysis using the time in hours to shut-down a C&C as the response variable. Our data consisted of overall C&C survival ($S(t)$), defined as the probability that a C&C is still online at a specific time during the study period. During that period, all C&Cs were initially online and may either have gone offline. Survival curves are estimated according to the Kaplan–Meier method [22], computed by updating the survival function after each event, without the assumption of an underlying probability distribution. Differences between survival curves were examined using a formal non-parametric statistical test called log-rank test [29]. It is worth noting that some C&Cs remain online at the end our study, making it impossible to observe when these are shut-down. C&Cs that remain online and in the search results at the end of our study are said to be right-censored.

We identify six different factors that might affect the lifetime of a C&Cs: location, malware variant, notification regime, hosting provider, registrar and popularity.

5.1 Location

First we analyze the impact of the location of the C&Cs on their lifetime. Fig. 4 shows the heat map of the average uptime of the C&C. Comparing the number of C&Cs hosted in each country to the average uptime of these C&C, we can observe a divergent pattern. While most of the C&Cs are hosted in the North America (41.29%), Europe (44.09%) and some populated countries in Asia (10.35%) (China and India), the worst countries in terms of average uptime are located in Africa, South America and China.

Fig. 5 shows the survival curves for the top 10 countries in terms of amount of C&Cs. We can appreciate that after the first 24 hours, the amount of C&Cs still online ranges from 78% to 85%. That indicates, that no matter the country, most of the C&Cs are alive for more than one day. Remarkably, the worst countries in terms of taking efficient measures against C&Cs are China, Canada and Russia. On the other hand, France and Netherlands are the countries that took C&Cs down faster.

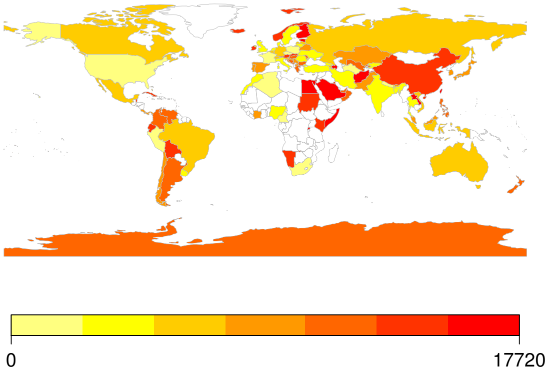


Figure 4: Country per C&C avg. uptime (hours)

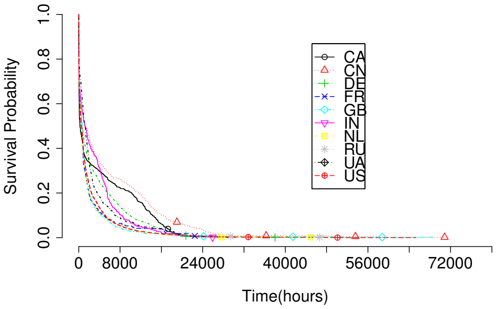


Figure 5: Kaplan-Meier estimates top 10 countries

To quantify the differences among different countries we perform a pairwise log-rank test[29]. Fig. 6 shows the results of the pairwise log-rank test for the top 20 countries in terms of amount of C&Cs. Blank tiles represent non-significant differences for a 5% confidence level. As expected, China and Russia survival curves are completely different to any other country. However, France survival rate is similar to the US, Romania, Netherlands, Macedonia, UK and Italy.

If we further analyzed the countries hosting ZeuS C&Cs with larger uptime, we notice that those countries have either lack of political stability, ineffective governance with high rates of terrorism or agents don't abide by the rules of society. Table 7 shows the top countries in terms of uptime the C&Cs along with different governance indicators

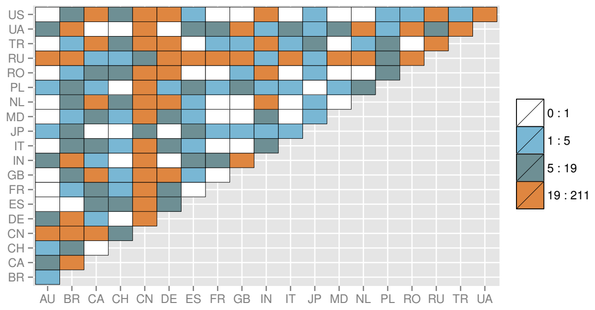


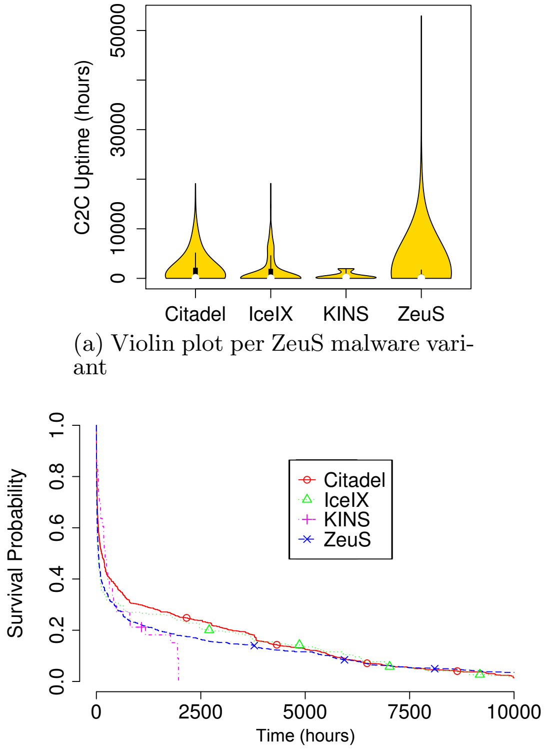
Figure 6: Differences between survival rates per country

[23]. Analyzing those indicators, we observe that the majority of these countries are below 1 which indicates political instability.

5.2 Malware variant

ZeuS malware has many variants among which the most prevalent ones are: Citadel, KINS, Ice-IX and the original ZeuS. All these variants have a similar purpose but the underneath communication protocol differs among them.

Fig. 7a shows the violin plot of the uptime for the 4 variants. We can appreciate differences both in terms of different moments as well as in the kernel density distribution of the uptime.



(a) Violin plot per ZeuS malware variant

(b) Kaplan-Meier estimates per malware variant

Figure 7: C&C Uptime per ZeuS malware variant

ZeuS is the variant that lives longer, while KINS has the lowest average uptime. Then we analyzed the survival rate for each of the variants. Fig. 7b show the Kaplan-Meier estimates per variant. As can be seen, Citadel presents the highest survival rate while KINS presents the lowest. On the one hand, this might be due to the Citadel's anti-reverse engineering techniques which hinder the malware analysis process. On the other hand, KINS presents a vulnerability in the file upload process that makes it straightforward to take it over.

To measure the differences among the survival rate per ZeuS variant, we perform a pairwise log-rank test. Table 8 shows the results of these tests. As can be seen, all the curves are significantly different from each other. Citadel survival curve presents the biggest difference when compared to ZeuS survival curve, while is not that different from IceIX survival curve.

Malware	ZeuS		Citadel		IceIX		KINS	
	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value
ZeuS			52.5	$4.27 \cdot 10^{-13}$	18.7	$1.5 \cdot 10^{-5}$	3.2	$7.3 \cdot 10^{-2}$
Citadel	52.5	$4.27 \cdot 10^{-13}$			3.5	$6.1 \cdot 10^{-2}$	19.1	$1.23 \cdot 10^{-5}$
IceIX	18.7	$1.5 \cdot 10^{-5}$	3.5	$6.1 \cdot 10^{-2}$			24.1	$9.34 \cdot 10^{-7}$
KINS	3.2	$7.3 \cdot 10^{-2}$	19.1	$1.23 \cdot 10^{-5}$	24.1	$9.34 \cdot 10^{-7}$		

Table 8: Log-rank test results per ZeuS variant

5.3 Notification regime

We further analyze the differences in terms of uptime for each of the data sources. Each data source has a different method to notify the URL of the detect C&Cs. ZeuS tracker presents a dynamic webpage that shows the online/offline C&Cs and also issues site summary and filter rules. Similarly, Cybercrime tracker also presents a dynamic webpage but they do not issue any type of periodic report/summary. Instead, Cybercrime tracker feeds one of the most prominent sources in terms of blacklisting checking as it is Virus Total [3]. On the other hand. the private company does not publicize any of the detected C&Cs. They only notify their customers about their existence so that they can take preventive measures.

We analyze how the different forms of notification affect the survival time of the C&Cs. Just comparing the average uptime we can already appreciate significant differences. C&Cs detected by ZeuS tracker stayed online an average of 1,328.64 hours, while the ones detected by Cybercrime tracker lived 22% less time. On the other hand, the C&Cs detected by the private company lived 2.8 times longer than the ones from ZeuS tracker.

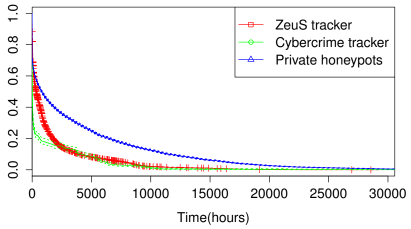


Figure 8: Uptime per source type

Then, we compare the different Kaplan-Meier estimates per data source. As expected, the C&Cs disclosed by Cybercrime tracker are the ones with lowest survival rates as they are included in the VirusTotal database that it is integrated by many companies. On the other hand, C&Cs detected by

Country	Political Stability	Gov. Effectiveness	Rule of Law	Uptime (hours)		
				Mean	Median	75th pctl
Virgin Islands (U.S.)	0.95	1.27	0.88	14143.00	14143.00	14143.00
St. Vincent&Grenadines	0.92	0.90	0.86	12557.00	12557.00	13198.50
Lao Pdr	0.06	-0.76	-0.77	10968.00	10968.00	12324.00
Somalia	-2.75	-2.21	-2.44	8841.00	8841.00	8841.00
Saudi Arabia	-0.41	0.06	0.26	8260.00	4138.00	12390.00
Tuvalu	1.32	-0.65	0.49	6472.82	757.50	9055.25
Rwanda	-0.08	0.00	-0.15	6470.00	3018.00	10521.00
Grenada	0.42	0.27	0.16	6262.00	6262.00	1710.00
Kenya	-1.15	-0.49	-0.74	5927.25	2465.00	7006.75
Palau	1.10	-0.59	0.90	5678.65	56785.65	3087.25
Afghanistan	-2.47	-1.43	-1.67	5574.00	5574.00	5574.00
Azerbaijan	-0.41	-0.46	-0.67	5505.33	84.00	8258.00
China	-0.55	-0.03	-0.46	5373.43	1318.00	8508.00
Samoa	1.01	0.14	0.72	4975.00	4318.00	12390.00
Cayman Islands	1.06	1.21	0.89	4767.33	2465.00	7691.00
Mexico	-0.74	0.31	-0.58	4664.13	1267.00	5378.75
Canada	1.03	1.77	1.74	4611.66	301.00	6273.50
Taiwan, China	0.86	1.19	1.04	4483.27	949.00	6726.00

Table 7: Governance indicators for top ZeuS hosting countries in terms of uptime (-2.5 to 2.5)

the private security company lived longer as they were not advertised publicly.

Table 9 shows the values of thee pairwise log-rank tests. The results show that all the survival curves are significantly different from each other. In particular, the survival curve from the private honeypots is well-differentiated from the other sources.

Source	ZeuS Tracker		Cybercrime		Priv. Honeypots	
	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value
ZeuS Tracker			238	0	1,730	0
Cybercrime	238	0			664	0
Priv. Honeypots	1,730	0	664	0		

Table 9: Log-rank test results per data source

5.4 Hosting providers & Registrars

In this section we go a step further and we analyze the impact on the survival rate of the C&Cs of parties involved in the take-downs.

First we analyze the hosting providers. Identified C&Cs were hosted in 1,139 different hosting providers. Table 10 shows the top 10 hosting providers in terms of number of C&Cs. United Internet AG is the German Internet services company that hosted more ZeuS C&Cs during period of this study. In terms of average uptime, C&Cs located in Microsoft's hosting infrastructure lived longer[1].

To further understand these differences, the hosting provider are classified according to the hosting service they offer as in [5]:

- Bulletproof: is a service provided by some domain hosting or web hosting firms that allows their customer considerable leniency in the kinds of material they may upload and distribute,
- Hacked: these are sites owned by legitimate people/business but that were comprised at some point and hosted C&C,

Hosting provider	**# C&Cs**	**Avg. Uptime(h)**
United Internet AG	5,006	3,975.54
Sprint	4,608	2,619.27
Microsoft Corporation	2,000	4,303.35
GoDaddy Inc	734	1,253.10
Hostinger Group	715	1,233.26
Endurance International	696	1,197.93
directi.com	658	2,818.34
FHE3 GmbH	560	3,100.40
Hurricane Electric	510	7,417.81
Hetzner Online AG	482	1,321.39

Table 10: Top hosting providers hosting ZeuS C&Cs

- Free hosting: is a hosting service that is free, usually advertisement-supported,
- Fast-Flux: is a hosting method used to hide servers or content behind an almost dynamic domain name.

According to this classification, 8.99% of the C&Cs are BulletProof, 0.01% are Fast-Flux, 2.43% Free hosting C&Cs, 24.19% Hacked and the rest didn't fit any of these categories. Then, we analyze the C&C uptime per category. As expected, Bulletproof and Fast-Flux providers are the ones that allow C&Cs to live longer. In this sense, C&Cs hosted by Fast-Flux providers stayed online for 2265.85 hours in average, while Hacked and Free-Hosted C&Cs only stayed online for less than 900 hours in average. Fig. 9 shows the survival curves for the four different types of hosting. As can be seen, "Fast-Flux" and "Bulletproof" hosting present the highest survival rates while "Free Hosting" and "Hacked websites" are cleaned up faster.

Finally, we compare the different survival curves by performing pairwise log-rank tests. Results show that not all the survival curves are significantly different from each other. At a 5% significant level, only "Bulletproof" is different from "Hacked" and "Free-hosting" from "Hacked". The rest of the survival curves are not significantly different among each other.

Next, we analyze how different registrars act in front of C&Cs that are registered by them. Different C&Cs were registered by more 400 different registrars. Table 12 shows

[1] Note that Microsoft sinkholed ZeuS C&Cs in several occasions. Sinkholing might lead to inaccuracies in the uptime of the sinkholed domain names.

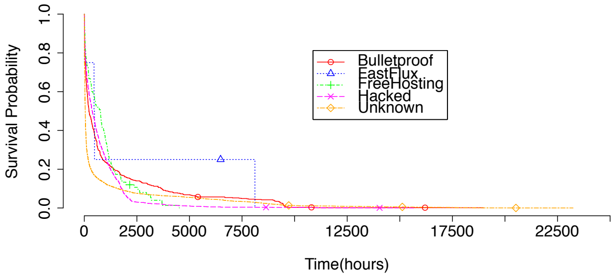


Figure 9: Kaplan-Meier estimates per hosting type

Hosting	Bulletproof		Hacked		Free-hosting		Fast-flux	
	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value	$\tilde{\chi}^2$	p-value
Bulletproof			8.2	0.004	0.4	0.514	0.2	0.657
Hacked	8.2	0.004			3.7	0.054	1.4	0.237
Freehosting	0.4	0.514	3.7	0.054			1.0	0.319
Fast-flux	0.2	0.657	1.4	0.237	1.0	0.319		

Table 11: Log-rank test results per hosting type

top 10 registrars in terms of number of C&Cs. There is a clear relationship between rogue registrars and rogue hosting providers. For instance, 1&1 Internet AG is the main registrar in terms of number of C&Cs. This same registrar registered 84.85% of the domains hosted by United Internet AG. However, Sprint hosting provider has domains registered by more than 40 different registrars including 1&1 Internet AG (3.36%), GoDaddy.com (5.04%), ENOM inc. (11.34%), Public domain registry (3.89%), Bizcn.com (26.31%), Internetbs corp (8.06 %) and Markmonitor (10.86%).

Registrar	**# C&Cs**	**Avg. Uptime(h)**
1&1 Internet AG	1793	5184.15
GoDaddy.com LLC	1018	2242.02
ENOM inc	972	2415.95
Public domain registry	796	1485.88
Bizcn.com inc	648	3091.97
r01-reg-ripn	471	830.95
Dnbiz limited	426	8052.21
Internetbs corp	333	3612.56
Markmonitor inc	288	2789.80
ru-center-reg-ripn	268	2235.01

Table 12: Top 10 registrars

To check if different C&Cs present different survival rate depending on the registrar they used, we estimate the Kaplan-Meier curves. From Fig. 10 we can see that C&Cs under registrars in Europe present lower survival rates than those registered by Asiatic or American registrars.

Finally, we perform pairwise log-rank tests to compare the different survival curves. Fig. 11 shows the value of the χ^2 test. We can observe that different registrars perform quite different in terms of C&C survival rate.

5.5 Popularity

We use Google's PageRank [25] as a measure of the popularity of the URLs hosting C&Cs. We analyze Google's PageRank as one of the factors that influence the lifetime of the C&Cs. Google's PageRank is a variant of the eigenvector centrality measure used commonly in network analysis that basically counts link votes and determines which URLs are most important. Results show that 66.94% of the C&Cs were ranked as 'rank 0', 11.48% 'rank 1', 10.02% 'rank 2', 7.02% 'rank 3', 3.22% 'rank 4', 1.04 'rank 5', 0.17% 'rank 6', 0.03% 'rank 7' and 0.06% 'rank 8'. As expected, most of the URLs hosting C&Cs were not very popular.

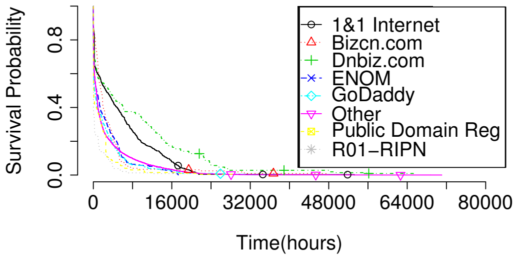


Figure 10: Kaplan-Meier estimates per registrar

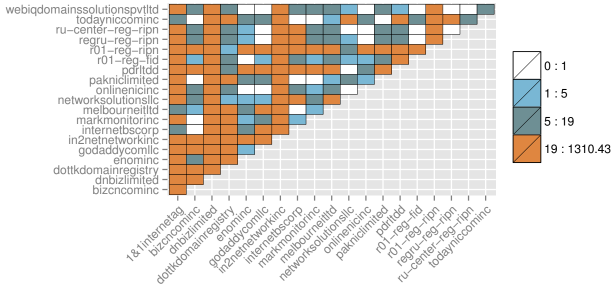


Figure 11: Pairwise log-rank test results per source

Fig. 12 presents the average C&C uptime depending on the rank of the URL where the C&C was hosted. C&Cs with a rank ranging from 0 to 4 have similar average uptime. However, C&Cs with ranks between 5-10 present an evident decreasing trend, i.e., the higher is the rank the lower is the uptime. Note that rank 7 C&Cs could be consider as an outlier as they only account for 0.03% of the number of C&Cs.

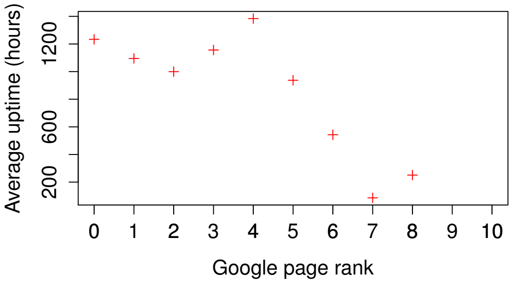


Figure 12: C&Cs avg. uptime depending on URL popularity

Finally, we compute the Kaplan-Meier estimates to compare the survival rates between C&Cs with different popularity. As can be seen in Fig. 13, there are significant differences among the difference curves. Rank 3 C&Cs live the longest while Rank 7/8 C&Cs are taken down in a matter of few hours.

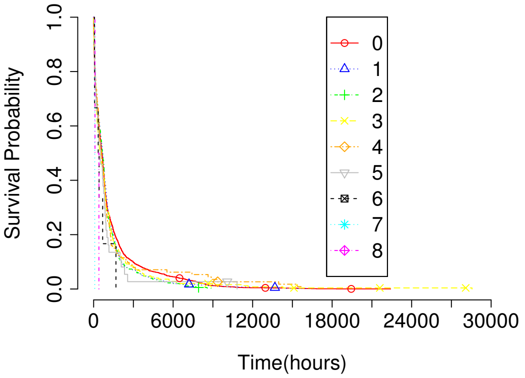


Figure 13: Kaplan-Meier estimates per domain popularity

6. STATISTICAL MODEL

To disentangle the effects of different demographic, technical, and social variables that could be associated with the life-cycle of ZeuS C&Cs, we use a Multivariate Cox proportional hazard model [14] of the form:

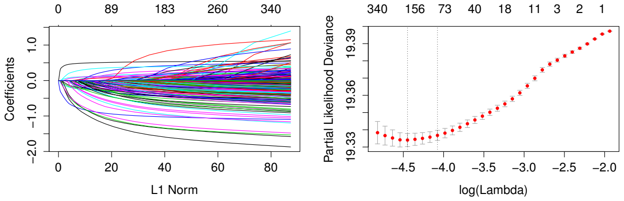
$$h(t) = h_0(t)e^{\beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m} = h_0(t)e^{\beta^{\mathbf{T}}\mathbf{x}} = \qquad (3)$$
$$= h_0(t)\exp(\beta_{1i}\text{Country}_i + \beta_{2i}\text{Family}_i +$$
$$+ \beta_{3i}\text{AS}_i + \beta_{4i}\text{Rank}_i + \beta_{4i}\text{Hosting}_i),$$

where $h(t)$ is the hazard for C&C i at time t, $h_0(t)$ represents the baseline hazard and the factor $\exp(\beta_i x_i)$ describes the hazard for a particular covariate x_i. The covariates correspond to the variables described in the previous section.

Cox model establishes a mechanism which allows survival time varies with hazardous factors [14]. Thus we can find out which factors have significant impact on the lifetime of the C&Cs and forecast the survival probability according to the influence of factors. The dependent variable in Cox model is hazard probability at a given time t.

Traditional factor selection techniques are based on statistical test like Chi–Square or Information criteria to estimate the importance of each factor independently. However, in the previous section we have already shown that different factors are highly correlated, e.g. location and hosting providers. Thus, we need an algorithm that considers correlations between factors so that we detect and delete redundant factors effectively. To that end, we use Lasso (Least Absolute Shrinkage and Selection Operator) [36] to fit a regularized regression model to factors in such a way that the final model is a sparse solution in the factor space. Hence, the weight of redundant features in the final model would be zero. This means that we can remove these features from the model with no significant change in the C&C lifetime prediction. Lasso has a regularization penalty parameter, λ, that offers a trade-off between sparsity of the model and the accuracy of the prediction. Lower values for λ result in more relaxation of the regularization constraint which allows more features to have non–zero weights.

Fig. 14a shows the value of the coefficients as a function of λ. Each curve corresponds to a different factor. It shows the path of its coefficient against the ℓ1-norm of the whole coefficient vector at as λ varies. The axis above indicates the number of nonzero coefficients at the current λ, which is the effective degrees of freedom for the Lasso. Fig. 14b shows the cross-validation curve (red dotted line), and upper and lower standard deviation curves along the λ sequence (error bars). The vertical dotted lines represent the values of λ that give minimum mean cross-validated error and the most regularized model such that error is within one standard error of the minimum respectively. As can be seen to obtain the minimum mean cross-validated error we have to include 180 factors.



(a) Coefficients

(b) cross-validation curve

Figure 14: Lasso model selection

λ	**Selected Features**
0.1442	Country={CA}

0.1314	Country={CA}
	Hosting={Hacked}

0.0825	Country={CA, RU}
	Hosting={Hacked}

0.0685	Country={CA, RU, AE}
	Hosting={Hacked}
	Family={ZeuS}
	ASN={33517, 50544}

...	...

0.0116	Country={AR, AZ, BY, BE, BR, BG, CA, KY, CN, CZ, EG, FR, HK, KZ, LV, LT, MT, MX, MD, PH, PT, RO, RU, RW, SK, TW, TH, UA, UK, VG}
	Hosting={Hacked}
	Family={ZeuS, Citadel, Ice, KINS}
	ASN={10029, 10474, 1101, 11340, 12140, 12322, 12406, 12481, 12573, 1267, 12849, 13147, 13193, 13237, 133165, 13335, 14037, 14080, 15497, 15830, 15967, 16125, 16257, 16509, 174, 18779, 19024, 19324, 19373, 19551, 19557, 196728, 199079, 19969, 20718, 20848, 20960, 21219, 24085, 24560, 24651, 25151, 25260, 25504, 25926, 26230, 26347, 26496, 278, 28824, 28917, 29290, 29761, 30083, 31103, 31133, 31147, 31252, 31727, 3249, 32592, 32613, 32780, 33317, 33517, 3352, 33922, 34011, 34187, 34702, 34788, 34977, 35088, 35908, 3598, 36351, 38370, 39020, 39647, 39756, 39779, 40244, 41175, 41310, 41786, 41794, 4250, 42668, 42831, 43060, 44018, 45324, 45887, 46062, 46816, 46844, 47242, 47544, 4765, 48031, 48172, 48185, 49126, 50274, 5033, 50544, 51248, 51743, 54301, 54456, 54718, 55470, 5588, 5617, 56309, 57043, 57910, 57972, 59491, 60708, 61214, 61255, 62416, 62567, 6429, 6903, 7381, 766, 790, 7922, 8001, 8075, 8426, 8447, 8560, 8828, 8926, 9070, 9121, 9198, 9269, 9891, 9916, 9918}
	Rank={0}

Table 13: Feature Selection using Lasso

Once fit, we can view the optimal λ value and a cross validated error plot to help evaluate our model. Table 13 shows a correlation between the selected factors and different values of λ. For $\lambda = 0.1442$, only one factor is selected. This means that, according to Lasso, if we want to forecast the lifetime of a C&C based on only one factor, checking where it is located would be our best choice. As we decrease the value of λ, more factors are selected according to their importance. The second most important factor is the type of hosting, specifically if the C&Cs is hosted in a hacked server. The last column in Table 13 shows the coefficients for λ that give the minimum mean cross-validated error. In this sparse factor set we have factors from all groups. For example,"ZeuS", "Citadel", "IceIX" and "KINS" are selected from the *Family* group. If we follow the Lasso path, factors get added and eliminated as λ decreases. We have a total of 180 factor selected from the 1,305 variables. In particular, we select 30 countries out of the 135 countries where C&Cs where located, we select 144 ASes out of the 1,153 ASes where C&Cs where located, we select all ZeuS family types, only "hacked" as hosting type and Google rank 0.

Then, we assess the goodness-of-fit of our model. Table 14 shows the values of various statistics. We calculate three different pseudo-R^2 parameters. All of them present low values which evidences that the impact of the factors is statistically significant but weak. Similarly, the concordance index has a value > 0.5 that implies that our model has good prediction capabilities.

	pseudo-$\mathbf{R^2}$			
AIC	**McFadden**	**Cox & Snell**	**Nagelkerke**	**C-Index**
39231.74	0.022	0.254	0.254	0.661

Table 14: Goodness-of-fit of the model

Finally, we determine whether the fitted Cox regression model adequately describes the data. To that end, we consider the violation of the assumption of proportional hazards. To validate the proportional hazard assumption (PHA) we calculate the scaled Schoenfeld residuals [18]. These residuals can be calculated as:

$$\hat{\vec{r}}_i = \vec{x}_i - E[\vec{x}_i], \tag{4}$$

we compute the expected values at time t_i as:

$$\begin{aligned} E[\vec{x}_i] &= \sum_{k \in \mathcal{R}(t_i)} \vec{x}_k p(k \text{ dies at } t_i) \\ &= \frac{\sum_{k \in \mathcal{R}(t_i)} \vec{x}_k e^{\vec{\beta}\vec{x}_k}}{\sum_{j \in \mathcal{R}(t_i)} e^{\vec{\beta}\vec{x}_j}}, \end{aligned} \tag{5}$$

where the values of $\vec{\beta}$ are the fitted coefficients, and $\mathcal{R}(t)$ is the set of C&Cs that are still online at time t.

The scaled Schoenfeld residuals are:

$$\hat{\vec{r}}_i^{*} = [Var(\hat{\vec{r}}_i)]^{-1}\hat{\vec{r}}_i \approx m\,Var(\hat{\vec{\beta}})\hat{\vec{r}}_i, \tag{6}$$

where *Var* denotes a covariance matrix, and m is the number of uncensored survival times. $\hat{\vec{r}}_i$ is a length-n vector, and $Var(\hat{\vec{\beta}})$ is a $n \times n$ matrix, where m is the number of uncensored data points and n is the number of features. These residuals are centered on zero and should be independent of time if PHA is true. Deviations from this, i.e. residuals that exhibit some trend in time, indicate that the PHA is violated. The Schoenfeld residuals, at the time when a failure or death were to occur, are defined by the difference between the observed and expected covariate values at that time [18].

We compute the scaled Schoenfeld residuals for each factors and perform a chi-square test between the Kaplan-Meier transformed survival times and the Schoenfeld residuals. Fig. 15 represents the two-sided p-value for each χ^2 test run over the Schoenfeld residuals versus the survival function. Note that all the p-values are above 0.2 which proves that there is no evidence to reject the hypothesis that the residuals are uncorrelated with time.

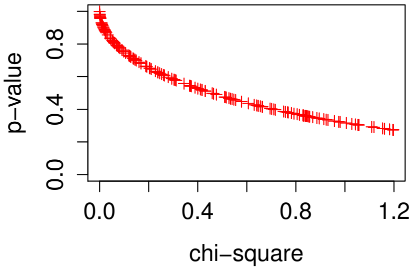


Figure 15: p-value of the χ^2 test over the scaled Schoenfeld residuals

7. RELATED WORK

While research has studied botnet identification and performance, no focus has been put into analyzing the factors that impact the lifetime of a command and control.

Rodríguez-Gómez et al. in [31] analyze the botnet life-cycle, understood as the sequence of stages that a botnet should successfully traverse in order to reach the success. They describe different defense mechanisms that affect the botnet during the different stages of its life-cycle. Apart from identifying these different stages, authors do not quantify the duration of each stage and do not identify the parties that are involved in the defense mechanism nor their role.

Researchers have also modeled the topology of different bot nets to gain insights into their life-cycle using deterministic [43, 42, 6, 20] and stochastic models [39, 26, 40, 41]. Recently, Khosroshahy et al. [24] developed a Markov-chain model that captures the dynamics of a botnet's life-cycle. However, the utility of the model is completely theoretical and it only allows to obtain expressions for the time dependent mean and variance of the population estimated size in each stage.

The rest of the literature is mainly focused on studying certain technical aspects like type of architecture, protocols or detection techniques [15, 27, 17, 19, 9, 8]. Different taxonomies have been proposed to classify botnets depending on their characteristics [16, 10, 33, 32].

Particularly, ZeuS botnet has also captured a lot of attention in the literature. The importance of ZeuS botnet is such that researchers have developed a tool for setting up and analyzing Zeus [21]. Stone-Gross [34] analyze the P2P variant of ZeuS in which disrupting the infrastructure (especially through legal action) becomes more difficult. Tajalizadehkhoob et al. [35] analyze the characteristics of the targets of ZeuS malware. Authors show that the critical bottleneck are the money mules, i.e., the accomplices to accept fraudulent ACH transfers who usually need to be within the same area or country as the victim to complete the transaction and reduce the risk of detection. Mohaisen et al. [28]

classify different version of ZeuS malware using several machine learning techniques using a behavior-based approach. They are able to identify 65 features that are unique and robust for identifying the different versions. Other research efforts have targeted at breaking ZeuS communication protocol. Ricardi et al. [30] propose a technique to break the encrypted malware communications, extracting the keystream used to encrypt such communications. Andriesse et al. [7] perform an in-depth analysis of the resilience Zeus botnet from the peer-to-peer perspective. Authors dissected the last version of the Zeus protocol describing the algorithm used and the resilient features applied. However, none of the previous research works have analyzed the factors that influence the lifetime of a ZeuS C&C.

8. CONCLUSIONS

This paper presents the first approach towards identifying the factors that affect the online time of the ZeuS C&Cs. Different technological characteristics are identified among the URLs where the C&Cs were hosted. Results showed some patterns that differ from the global market view. ZeuS C&Cs remained online for longer periods when hosted in some low usage web servers like Tengine or when using non-popular CMSes.

Furthermore, we proved from the Kaplan-Meier survival estimates there is a significant difference in survival by the location, notification regime, malware family, hosting provider, registrar and popularity of the C&Cs. On the other hand, the logrank test also showed that there was no significant difference in survival experience between the various categories of some covariates.

Multivariable Cox hazards regression model revealed that location, malware variant, AS number, hosting type and popularity were significant factors associated with the online time of the C&Cs. Via a penalized likelihood analysis we detected those categories with a major impact on the lifetime of ZeuS C&Cs. Our model presents good prediction capabilities (C-index 0.66).

Overall, our findings have important implications for information systems policy as well as for implementation of botnet defense techniques for ensuring the safety of network systems. Our prediction model can be use to estimate the lifetime of a given C&C and detect which factors represent the main hazard.

9. ACKNOWLEDGMENTS

The authors would like to thank Roman Huessy for sharing ZeuS C&C data. Also, thanks to TU Delft's ICT and Abuse staff (Lolke Boonstra, Adrian Cooke, and Paul Keekstra), SUFRNet (Wim Biemolt) for their support. This work was partly funded by the Netherlands Organisation for Scientific Research, grant number 628.001.022.

10. REFERENCES

[1] Cybercrime-Tracker. http://cybercrime-tracker.net/, Sept 2014.

[2] Netcraft Toolbar. http://toolbar.netcraft.com/, Oct 2014.

[3] Virus Total. https://www.virustotal.com/, Oct 2014.

[4] W3Techs- Web Technology Surveys. http://w3techs.com/reports, Oct 2014.

[5] ZeuS Tracker. https://zeustracker.abuse.ch, Sept 2014.

[6] M. Ajelli, R. Lo Cigno, and A. Montresor. Modeling botnets and epidemic malware. In *Communications (ICC), 2010 IEEE International Conference on*, pages 1–5, May 2010.

[7] D. Andriesse, C. Rossow, B. Stone-Gross, D. Plohmann, and H. Bos. Highly resilient peer-to-peer botnets are here: An analysis of Gameover Zeus. In *Malicious and Unwanted Software: "The Americas" (MALWARE), 2013 8th International Conference on*, pages 116–123, Oct 2013.

[8] M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou, II, and D. Dagon. Detecting malware domains at the upper dns hierarchy. In *Proceedings of the 20th USENIX Conference on Security*, SEC'11, pages 27–27, 2011.

[9] 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 *Proceedings of the 21st USENIX Conference on Security Symposium*, Security'12, pages 24–24, 2012.

[10] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir. A survey of botnet technology and defenses. In *Conference For Homeland Security, 2009. CATCH '09. Cybersecurity Applications Technology*, pages 299–304, March 2009.

[11] 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 Eighth Annual International Conference on*, pages 31–38, Aug 2010.

[12] G. Box and G. Tiao. Intervention analysis with applications to economic and environmental problems. *J. Am. Stat. Assoc.*, 70:70–79, 1975.

[13] D. Brown. Resilient botnet command and control with tor. In *DEF CON 18 - Hacking conference*, 2010.

[14] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):pp. 187–220, 1972.

[15] A. Dainotti, A. King, k. Claffy, F. Papale, and A. Pescapè. Analysis of a "/0" stealth scan from a botnet. In *Proceedings of the 2012 ACM Conference on Internet Measurement Conference*, IMC '12, pages 1–14, 2012.

[16] N. Daswani and M. Stoppelman. The Anatomy of Clickbot.A. In *Proceedings of the First Conference on First Workshop on Hot Topics in Understanding Botnets*, HotBots'07, pages 11–11, Berkeley, CA, USA, 2007. USENIX Association.

[17] S. García, M. Grill, J. Stiborek, and A. Zunino. An empirical comparison of botnet detection methods. *Comput. Secur.*, 45:100–123, Sept. 2014.

[18] P. M. Grambsch and T. M. Therneau. Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81(3):515–526, 1994.

[19] F. Haddadi, D. Runkel, A. N. Zincir-Heywood, and M. I. Heywood. On Botnet Behaviour Analysis Using GP and C4.5. In *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation*

Companion, GECCO Comp '14, pages 1253–1260, 2014.

[20] Q. Han, W. Yu, Y. Zhang, and Z. Zhao. Modeling and evaluating of typical advanced peer-to-peer botnet. *Perform. Eval.*, 72:1–15, Feb. 2014.

[21] K. Hannah and S. Gianvecchio. Zeuslite: A Tool for Botnet Analysis in the Classroom. *J. Comput. Sci. Coll.*, 30(3):109–116, Jan. 2015.

[22] E. L. Kaplan and P. Meier. Nonparametric Estimation from Incomplete Observations. *Journal of the American Statistical Association*, 53(282):457–481, 1958.

[23] D. Kaufmann, A. Kraay, and M. Mastruzzi. The worldwide governance indicators. The World Bank, `http://info.worldbank.org/governance`, 2014.

[24] M. Khosroshahy, M. K. Mehmet Ali, and D. Qiu. The SIC Botnet Lifecycle Model: A Step Beyond Traditional Epidemiological Models. *Comput. Netw.*, 57(2):404–421, Feb. 2013.

[25] A. N. Langville and C. D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, Princeton, NJ, USA, 2006.

[26] X. Li, H. Duan, W. Liu, and J. Wu. The growing model of botnets. In *Green Circuits and Systems (ICGCS), 2010 International Conference on*, pages 414–419, June 2010.

[27] M. M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham. Peer to peer botnet detection for cyber-security: A data mining approach. In *Proceedings of the 4th Annual Workshop on Cyber Security and Information Intelligence Research: Developing Strategies to Meet the Cyber Security and Information Intelligence Challenges Ahead*, CSIIRW '08, pages 39:1–39:2, 2008.

[28] A. Mohaisen and O. Alrawi. Unveiling Zeus: Automated Classification of Malware Samples. In *Proceedings of the 22Nd International Conference on World Wide Web Companion*, WWW '13 Companion, pages 829–832, 2013.

[29] R. Peto and J. Peto. Asymptotically efficient rank invariant test procedures. *Journal of the Royal Statistical Society. Series A (General)*, 135(2):pp. 185–207, 1972.

[30] M. Riccardi, R. Di Pietro, M. Palanques, and J. A. Vila. Titans' Revenge: Detecting Zeus via Its Own Flaws. *Comput. Netw.*, 57(2):422–435, Feb. 2013.

[31] R. A. Rodríguez-Gómez, G. Maciá-Fernández, and P. García-Teodoro. Survey and taxonomy of botnet research through life-cycle. *ACM Comput. Surv.*, 45(4):45:1–45:33, Aug. 2013.

[32] C. Rossow, D. Andriesse, T. Werner, B. Stone-Gross, D. Plohmann, C. Dietrich, and H. Bos. SoK: P2PWNED - Modeling and Evaluating the Resilience of Peer-to-Peer Botnets. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 97–111, May 2013.

[33] S. S. C. Silva, R. M. P. Silva, R. C. G. Pinto, and R. M. Salles. Botnets: A survey. *Comput. Netw.*, 57(2):378–403, Feb. 2013.

[34] B. Stone-Gross. The Lifecycle of Peer-to-Peer (Gameover) ZeuS. Dell SecureWorks Counter Threat Unit(TM) Threat Intelligence, 2012.

[35] S. Tajalizadehkhoob, H. Asghari, C. Gañán, and M. van Eeten. Why Them? Extracting Intelligence about Target Selection from ZeuS Financial Malware. In *13th Annual Workshop on Economics of Information Security (WEIS 2014)*, pages 1–26, 2014.

[36] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[37] H. Tiirmaa-Klaar, J. Gassen, E. Gerhards-Padilla, and P. Martini. *Botnets*. SpringerBriefs in Cybersecurity. Springer, 2013.

[38] M. van Eeten, J. M. Bauer, H. Asghari, S. Tabatabaie, and D. Rand. The Role of Internet Service Providers in Botnet Mitigation: An Empirical Analysis Based on Spam Data. In *9th Annual Workshop on the Economics of Information Security*, WEIS 2010, 2010.

[39] E. Van Ruitenbeek and W. Sanders. Modeling peer-to-peer botnets. In *Quantitative Evaluation of Systems, 2008. QEST '08. Fifth International Conference on*, pages 307–316, Sept 2008.

[40] Q. Wang, Z. Chen, C. Chen, and N. Pissinou. On the robustness of the botnet topology formed by worm infection. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6, Dec 2010.

[41] R. Weaver. A Probabilistic Population Study of the Conficker-C Botnet. In *Proceedings of the 11th International Conference on Passive and Active Measurement*, PAM'10, pages 181–190, 2010.

[42] W. Xin-liang, C. Lu-Ying, L. Fang, and L. Zhen-Ming. Analysis and modeling of the botnet propagation characteristics. In *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, pages 1–4, Sept 2010.

[43] C. C. Zou and R. Cunningham. Honeypot-aware advanced botnet construction and maintenance. In *Proceedings of the International Conference on Dependable Systems and Networks*, DSN '06, pages 199–208, 2006.