

# Understanding the Role of Sender Reputation in Abuse Reporting and Cleanup

Orcun Cetin\*, Mohammad Hanif Jhaveri<sup>†</sup>, Carlos Gañán\*, Michel van Eeten\*, Tyler Moore<sup>†</sup>

\*Delft University of Technology, Faculty of Technology, Policy and Management  
{f.o.cetin, c.h.g.hernandezganan, m.j.g.vaneeten}@tudelft.nl

<sup>†</sup>Southern Methodist University, Computer Science and Engineering Department  
{mjhaveri@alumni.smu.edu, tylerm@smu.edu}

**Abstract**—Participants on the front lines of abuse reporting have a variety of options to notify intermediaries and resource owners about abuse of their systems and services. These can include emails to personal messages to blacklists to machine-generated feeds. Recipients of these reports have to voluntarily act on this information. We know remarkably little about the factors that drive higher response rates to abuse reports. One such factor is the reputation of the sender. In this paper, we present the first randomized controlled experiment into sender reputation. We used a private datafeed of Asprox-infected websites to issue notifications from three senders with different reputations: an individual, a university and an established anti-malware organization. We find that our detailed abuse reports significantly increase cleanup rates. Surprisingly, we find no evidence that sender reputation improves cleanup. We do see that the evasiveness of the attacker in hiding compromise can substantially hamper cleanup efforts. Furthermore, we find that the minority of hosting providers who viewed our cleanup advice webpage were much more likely to remediate infections than those who did not, but that website owners who viewed the advice fared no better.

## I. INTRODUCTION

Advances in detecting and predicting malicious activity on the Internet, impressive as they are, tend to obscure a humbling question: Who is actually acting against these abusive resources? The reality is that the bulk of the fight against criminal activity depends critically on the voluntary actions of many thousands of providers and resource owners who receive abuse reports. These reports relay that a resource under their control – be it a machine, account, or service – has been observed in malicious activity. Each day, millions of abuse reports are sent out across the Internet via a variety of mechanisms, from personal messages to emails to public trackers to queryable blacklists with thousands of hacked sites or millions of spambots.

Proactive participants may pull data from clearinghouses such as Spamhaus and Shadowserver. But in many cases, the reports are pushed to recipients based upon publicly available abuse contact information. In these circumstances, those who can act against the abusive resource might never actually see the information. If the information does reach them, it might be ignored, misunderstood or assigned low priority. Still, against all these odds, many reports are acted upon, without any formal requirement, across different jurisdictions and often without a pre-established relationship between sender

and recipient. This voluntary action is an under-appreciated component of the fight against cybercrime.

Remarkably little research has been undertaken into what factors drive the chances of a recipient acting upon an abuse report (notable exceptions are [1]–[4]). One factor, the reputation of the sender, clearly plays an important role in practice. Not all reports are treated equal, as can be seen from the fact that some recipients assign a trusted status to some senders (‘trusted complainer’), sometimes tied to a specific API for receiving the report and even semi-automatically acting upon it.

The underlying issue is a signaling problem, and therefore, an economic one. There is no central authority that clears which notifications are valid and merit the attention of the intermediary or resource owner. This problem is exacerbated by the fact that many intermediaries receive thousands of reports each day. One way to triage this influx of requests for action is to judge the reputation of the sender.

We present the first randomized controlled experiment to measure the effect of sender reputation on cleanup rates and speed. During two campaigns over December 2014–February 2015, we sent out a total of 480 abuse reports to hosting providers and website owners from three senders with varying reputation signals. We compared their cleanup rates to each other and to a control group compromised with the same malware.

In the next section, we outline the experimental design. In Section III, we turn to the process of data collection, most notably tracking the cleanup of the compromised resources that were being reported on. The results of the experiment are discussed in Section IV. Surprisingly, we find no evidence that sender reputation improves cleanup. We find that the evasiveness of the attacker in hiding compromise can substantially hamper cleanup efforts. Furthermore, we find that the minority of hosting providers who viewed our cleanup advice were much more likely to remediate infections than those who did not, but that website owners who viewed the advice fared no better. We compare our findings to related work in the area in Section V. We describe limitations in Section VI and conclude in Section VII.

## II. EXPERIMENTAL DESIGN

Does sender reputation matter when notifying domain owners and their hosting providers with evidence that their website is compromised? We designed an experiment measuring cleanup rates as a result of abuse reports sent from three senders with varying levels of reputation: an unknown individual, a university and StopBadware, a well-established non-profit organization that fights malware in collaboration with industry partners.

The analysis and data collection started in December 2014 and continued through the first week of February 2015 across two campaigns. Figure 1 illustrates the rules we applied to get the experimental data set from the original feed.

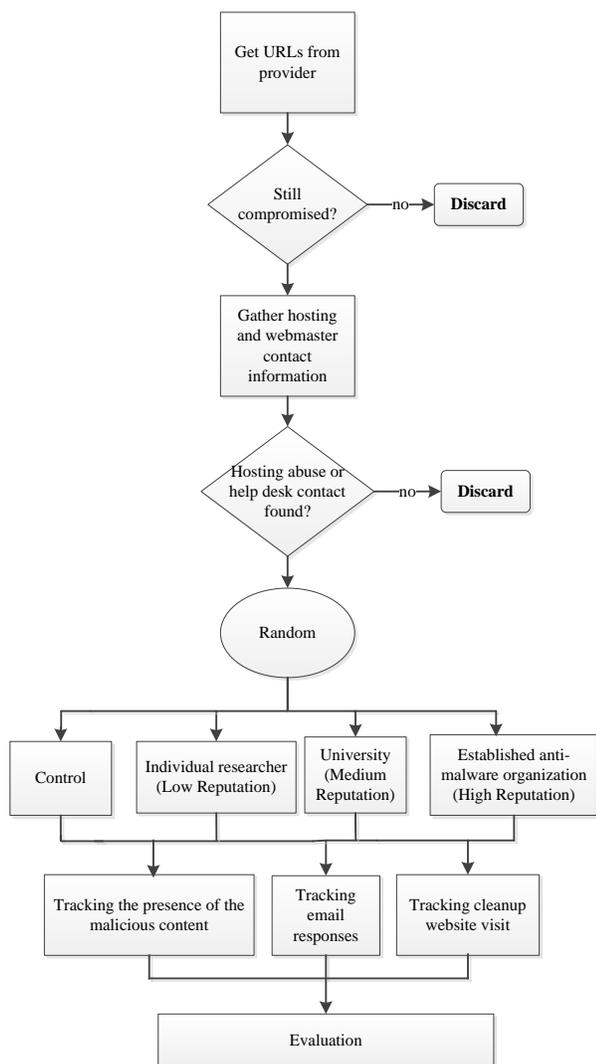


Fig. 1: Flow diagram of the progress through the phases of our experiment

### A. Study Population and Sampling

The study population was derived from a raw daily feed of URLs serving malicious downloads originating from the

Asprox botnet. This private source of abuse data was not shared with anyone else and free of any prior notification attempts.

From December 7th, 2014 until January 19th, 2015, we received a total of 7,013 infected URLs. We checked whether the site was indeed still compromised. In a handful of cases, cleanup or remediation seemed to have taken place already. If so, the URL was discarded. Next, we looked up abuse contact information for the hosting provider and the domain owner from WHOIS data. If we could not find any contact information for the hosting provider (for example, if the WHOIS information was set to private), we discarded the URL. When we did not find any contact information for the domain owner, we would use the RFC standard abuse e-mail address [5]. All in all, we discarded fewer than 10 URLs for either no longer being compromised or the lack of an abuse contact for the hosting provider.

From the remaining set, we took a random sample. This was done each day that new URLs were being supplied to us. The daily feed fluctuated dramatically, with peaks of close to one thousand URLs and days with just a handful. Most days, we received between 50-100 URLs. From these, we took a daily random sample, typically of around 40 URLs. We could not include all URLs we received in the experiment because of a bottleneck further on in the process: tracking the up-time of the compromised content (see Section III).

To determine the total sample size, in other words how many URLs we needed, we completed a power calculation for the main outcome variable, cleanup rate. We estimated power for three levels: 80%, 85% and 90% and used a 5.65 standard deviation based on prior studies [1]. Differences in mean sixteen-day cleanup time of about 0.84 days between conditions can be detected with 90% power in two-tailed tests with 95% confidence, based on a sample of 80 websites in each treatment group. To ensure that the control has enough statistical power for baseline comparison across treatment groups, we set the control equal to all other treatment groups combined. This resulted in a total sample size of 482 URLs.

### B. Treatment Groups & Rationale

Using a random number generator, we assigned URLs to a treatment condition or to the control group. The three treatment conditions were sending an abuse report from an individual researcher, a university and an established anti-malware organization (see Table I). The report from the individual researcher was designed to reflect a low reputation abuse notifier and was sent from a Gmail account. The university group was set up to reflect a medium reputation abuse notifier. Here, we used a functional e-mail address from Delft University of Technology. The established anti-malware organization was included as the sender with the highest reputation. StopBadware generously provided us an e-mail account at their domain to send notifications on their behalf [6].

As the randomization took place at a URL level, the domain owner and the hosting provider were assigned to the same

Group	Description	E-mail Address	Sample Size		Rationale
			Camp. 1	Camp. 2	
Control	No notifications	N/A	17	229	Baseline to understand the natural rate of compromised host survival
Individual researcher	Individual internet researcher	malwarereporting@gmail.com	23	57	Individuals may send mixed signals, from quality to motivation
University	Academic institution	malwarereporter-tbm@tudelft.nl	17	62	Academic organizations may signal higher quality and research intent
Established Anti-malware Organization	Anti-malware nonprofit organization	abuse-reporter@stopbadware.org	20	61	Dedicated organizations may signal the highest quality research and/or potential commercial enforcement

TABLE I: Overview of each treatment group

treatment group. The notified entities were, by nature of the intervention, not blinded.

Once assigned, we completed a statistical analysis on key attributes to ensure the assignments were comparable across groups. The control group served as a baseline to understand the natural survival rate of a compromise and was the only one not to receive notifications. There was no difference among the treatment groups other than the domain of the e-mail address and the host of the cleanup content. We base this on studies [7] that indicate users perceive domains with certain top-level extensions to have differing levels of authority in terms of the accuracy of information.

### C. Notification & Cleanup Support Site

The abuse notifications were based on the best practice for reporting malware URLs that has been developed by StopBadware [8]. The content included the malicious URL, a description of the Asprox malware, the IP address, date and time of the malware detection and a detailed description of the malware behavior. Abuse notification sample for established anti-malware organization, university and individual internet researcher are respectively presented in Appendix figure 11, 12 and 13.

We sent notifications to each treatment group during 12 days in total. All treatment groups received an identical abuse notification, except for the sender e-mail address and the included link to a web page where we described cleanup advice for sites compromised by Asprox. The web page provided a brief guide explaining how to identify and remove Asprox malware and backdoors from compromised websites. The page also included links to other websites for precautionary measures to prevent the site from being compromised again. Figure 14 in the Appendix, contains samples of the various cleanup websites shared in the e-mail notification for each of the treatment groups.

The webpage was hosted at different domains consistent with each treatment condition. The individual researcher e-mailed a link to a free hosting webpage, the university to a page inside the official TU Delft website, and StopBadware to a page on their official domain.

Furthermore, each cleanup link contained a unique seven-character code allowing us to track which recipients clicked on the link. In this way, we measure whether visiting the cleanup page was associated with higher cleanup rates.

To prevent biases because of the recipients' varying abilities to receive the e-mail and view the webpage, we tested all the e-mail notifications across various e-mail services to ensure correct delivery and double-checked that the webpages were not on any of the major blacklists.

### D. Evaluation

We evaluate the experiment based on the differences in cleanup rates and median-time to cleanup across the various treatment groups relative to the control group. We also explore the relationship between cleanup rates and other variables, such as visits to the cleanup advice page and the responses of providers to our notifications.

## III. DATA COLLECTION

To perform the experiment designed in the previous section, we received assistance from an individual participating in the working group analyzing and fighting the Asprox botnet. He supplied us with a private feed of URLs in use by Asprox. The URLs were captured via spamtraps and various honeypot servers located in Europe and the United States.

The Asprox botnet was first detected in 2007. Since then, it has evolved several times. Currently it is mostly used for spam, phishing, the distribution of malware to increase the size of its network, and for the delivery payload of pay-per-install affiliates [9]. Asprox compromises websites by building a target list of vulnerable domains and then injects SQL code that inserts a PHP script that will trigger the visitor to download malware or redirect them to various phishing sites. Our URL feed contained both variations.

### A. Evolution of Asprox compromised sites

In the course of our experiment, Asprox's behavior changed as it went through two different attack campaigns (see Table II). From December 2014 until beginning of January 2015, the infected sites delivered a malicious file. After that, from January 2015 until February 2015, instead of delivering a malicious file, infected domains redirected visitors to an ad-fraud related site. Moreover, these two campaigns did not only differ on the type of malicious behavior but also on the countermeasures taken by the botnet against detection and removal.

During the first campaign, the botnet's countermeasures included blacklisting of visitors to the compromised sites based

Campaigns	Start Date	End Date	Type	Character
Campaign 1	12/08/2014	12/26/2014	Malware	* Customized and standard error messages * IP and identifier based blacklisting
Campaign 2	01/12/2015	02/04/2015	Ad-fraud	* Standard error message

TABLE II: Overview of each campaign

on IP addresses and machine fingerprinting. The blacklist was managed by back-end command-and-control systems and shared among the compromised sites.

Once an IP address was blacklisted, the compromised sites stopped serving the malicious ZIP file to that particular IP and displayed an error message instead. We encountered two different types of error messages: (i) HTTP standard error messages such as 404 Not Found, and (ii) customized error messages such as “You have exceeded the maximum number of downloads”. In addition, sites only accepted requests coming from Internet Explorer 7 and versions above.

In contrast to the first campaign, the second campaign did not apply any type of blacklisting. Instead the main countermeasure consisted of displaying an error message when trying to access the malicious PHP file alone. Moreover, the path to reach the malicious content would change periodically.

In most cases, the malicious content was only accessible through the URLs included in the phishing e-mails. These URLs included a request code that allowed infected sites to serve malware binaries and phishing pages that belonged to a specific Asprox attack. Once that specific attack ended, the compromised sites stopped responding to the corresponding URLs and displayed an error message instead. Table III shows a list of request codes and the corresponding attributes for both malware and phishing URLs. For instance, “?pizza=” code was only used for triggering PizzaHut\_Coupon.exe Asprox malware binary.

### B. Tracking presence of malicious content

Given the evolution and countermeasures of the Asprox botnet, the experiment required a complex methodology to track the notified entities acted upon our abuse report and cleaned up the compromised site. In the following, we describe the notification process and the methodology to track Asprox infected websites.

To identify and monitor malicious content for the first campaign, we first required a mechanism to bypass the botnet’s blacklisting of visitors based on IP-addresses and fingerprinting. The compromised sites used error messages to make it harder to distinguish malicious links from broken or dead links. We developed an automated tool that used IP addresses from 2 private and 7 public HTTP proxy services and checked whether the IP address that the tracking tool received had not been used before. Each day, 3 different proxy services were selected. All new IP addresses were checked against a list of previously used IP addresses. If it has been previously used, we discarded it. If not, we added it to the list. The IP addresses were selected following a round-robin algorithm from the pool of proxy services.

During a 16-day tracking period, we followed the procedure outlined in Figure 2 to determine whether a site was considered to be clean or compromised. Exactly 16 of the 486 total compromised sites (3%) periodically did not resolve. All were from the second campaign: 10 in the control group, 4 in the established anti-malware organization group, and 2 in the individual researcher group. While this might imply the site has been cleaned, that isn’t always the case. Earlier work indicates that clean-up actions are sometimes visible in the WHOIS data [1], specifically in the status fields. We identified three cases (two in established anti-malware organization group and one in individual researcher group) where the Domain Status and other fields of the WHOIS records changed, indicating that content of the site was removed. In the other 13 cases, we had no clues to clearly determine whether the site was actually cleaned up or in temporarily maintenance. Thus, we considered these 13 cases still infected.

Finally, in situations where the domain name resolved but the URL returned an HTTP error code different from HTTP 404 (Not Found), we also assumed that the malicious file was still present.

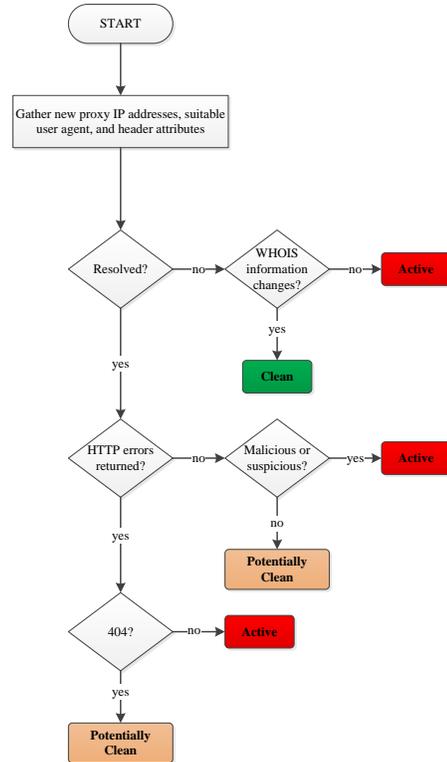


Fig. 2: Flow chart for following up to determine when clean

Malware Campaign			
Request Code	Targeted Companies	Sample	Name of Executable
?c=	Costco	?c=r24t/fwI8nYJeoktSMii3IkC8ItN3Dqcpphcm375Sg4	Costco_OrderID.exe
?fb=	Facebook	?fb=i2uXy5/kOZ77bjvMAA0hgsai4YbZNVc78Ji7amd1D8Y	FB-Password-Reset_Form.exe
?w=	Walgreens	?w=uhUGpftxxueBCfO/6FxAx7p2/Guz9BjRwRj/1YVMcKI	Walgreens_OrderID.exe
?pizza=	Pizza Hut	?pizza=Wa5wEaLOSojFl3kTaW3OIgOW150DCm7Jda8m83pzVJo	PizzaHut_Coupon.exe

Ad Fraud and Phishing Campaign		
Request Code	Type of Scam	Sample
?po=	Ad-Fraud	?po=rldsS+cFDm7bNp4duz57G0IWqGTH15cqcKUdvtSGBME
?r=	Dating Website Scam	?r=2

TABLE III: Examples request codes and what they represent.

When a server successfully returned some content or a redirection to another website, our scanner analyzed the content searching for common Asprox malicious behavior. This procedure is summarized in Figure 3.

In both campaigns, we started by accessing the infected website and analyzing the HTTP server header request. If the server returned HTTP 200 (OK), then we further analyzed the header’s content-disposition field to assess the attachment of a file with a .zip extension, which would contain the malicious binaries. If the website delivered a zip file, we concluded that the malicious script was still present and the website remained compromised.

The absence of an attachment in the website did not necessarily indicate that the site was clean. In some cases, infected sites were acting as redirectors to various phishing and ad-fraud sites. To capture this behavior, we analyzed the HTML content of the infected websites looking for a specific combination of HTML tags that were used for redirecting to known ad-fraud and rogue pharmacy sites that were captured during previous scans. If the redirected site led to malicious content we marked it as being compromised.

When clearly malicious content was not present in redirected site, we manually entered it into the VirusTotal [10] website query field. We then selected “Re-Analyze” to ensure that the checker was being run at the point of our query to have the service return whether the site was currently blacklisted or not. When the site returned that the URL or domain was in the blacklist, we marked it as being malicious. When indicated as being clear, we followed up and ran it through a passive DNS replication service to see if the resolved IP address hosted any other Asprox-related site. If found, we concluded that the site was still compromised.

We also inspected the HTML content associated with PHP fatal errors, disabled, and suspended sites. Disabled and suspended pages might indicate that action was taken to mitigate the abuse, even though the malicious script might still remain. In two cases, malicious links displayed a PHP fatal error [11]. While this could be related to a programming error, the ones we reviewed included HTML tags that are specifically associated with malicious content. Hence we assume that this

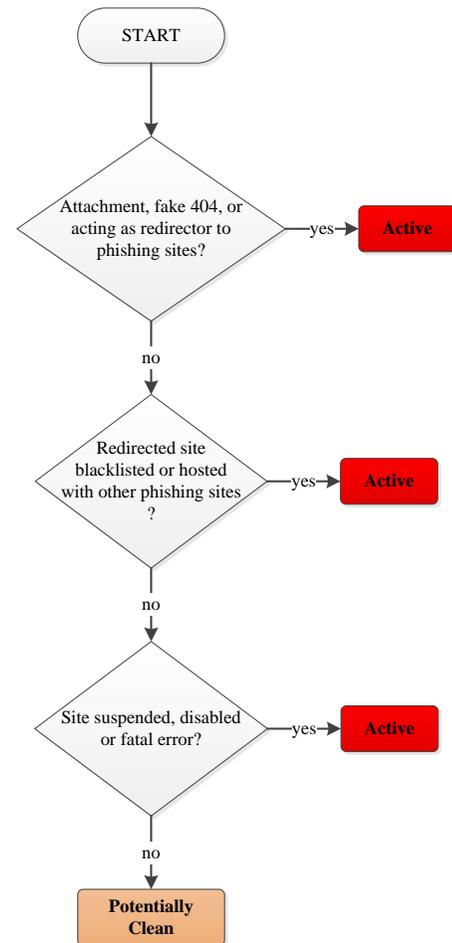


Fig. 3: Flow chart for deciding whether a site is malicious

implied the site was still compromised, and possibly just temporarily generating the fatal error to hide from hosting provider clean-up efforts.

When the website returned a HTTP 404 (Not Found) error message or in the absence of a clear indicator of malicious content, we classified the compromised site as potentially clean

since the botnet infrastructure had modules to prevent security bots from reaching the malicious content. To gather more information about these potentially clean websites, we scan those sites 2 more times on the same day. If during these 2 additional scans no indicators of malicious or suspicious behavior were found, follow-ups scans were performed during the next 2 days with 3 unique requests. If there was no malicious or suspicious behavior during 3 consecutive days, then we considered the site to be potentially clean and manually investigated the URLs using online server header checker websites (e.g. [12]) and by visiting it manually using a 'clean' set of IP addresses that were acquired via a premium VPN subscription. These manual follow-ups were made to ensure reliable measurements on the presence of malicious content. The evolution of Asprox made it impossible to fully rely on automation. In the end, we only considered a site clean if it was never subsequently observed to be malicious in manual and automated scans.

During the second campaign, the botnet infrastructure was no longer using blacklisting based on IP addresses or fingerprinting. Therefore, we only used IP addresses from a single HTTP proxy service to track the presence of malicious content. As a preventive measure, our scanner used a mechanism where IP addresses were changed twice a day and different browser suits were used to visit the site. Only one followup was made for each day of tracking due to lack of blacklisting. Another difference with the first campaign was that scans for the last day of tracking was automated. We only considered a site clean if, and only if, there was no malicious content related to Asprox botnet in both followups and last day scans.

Throughout the tracking process of the second campaign, compromised sites stopped redirecting to ad-fraud sites and paths to ad-fraud campaign were displaying standard error messages. This indicated that Asprox ad-fraud campaign was over. New links were generated by the botmasters for redirecting to the new scams sites such as fake dating or diet websites. Thus, the same infected websites that were used during the second campaign to redirect to ad-fraud related websites were now being used to redirect to other type of scams.

### C. Tracking affected party responses

As part of the experiment, we also regularly checked the inbox of the different e-mail accounts created for this study. We received automated and manual responses from the affected parties. Automated responses came from hosting providers to acknowledge the reception of our notification. Most of the automated responses contained a ticket number, to be included in further communication about the infection. Some providers also included details of the ticket along with a URL for tracking the incident status.

Manual responses came from domain owners and abuse-desk employees to inform us about the cleanup action taken or requesting more evidence about the compromise. When we received a manual response stating that appropriate action was taken, we re-scan the website to confirm this action. If the results of the scan found that the infection was still present, we responded to the corresponding entity stating the existence

of the malicious PHP script. In these responses, a HTTP header request from the malicious URL was included to serve as evidence showing the existence of the malicious file. When more evidence of the compromise was requested, a brief explanation of the compromise and a specific solution was given.

We also analyzed the logs of our web pages with cleanup advice. Via the unique codes included in the URLs, we identified which hosting provider or site owner visited one of our cleanup websites. Unfortunately, we discovered in the course of the experiment that the server logs for the StopBadware page could not be analyzed, as the webserver relied on Cloudflare's CDN service to serve the static content, thus leaving no log of the visit [13].

## IV. RESULTS

From December 7th, 2014 until January 19th, 2015, a total of 7,013 infected URLs were identified. From these we excluded less than 10 URLs that were not active or for which we were not able to obtain reliable contact information for the hosting provider. The daily feed fluctuated dramatically, with peaks of close to one thousand URLs and days with just a handful. Most days, we received between 50-100 URLs. From these, we took a daily random sample, typically around 40. Over time, this accumulated to a random sample of 486 URLs.

In the following we empirically estimate the survival probabilities using the Kaplan-Meier method. Survival functions measure the fraction of URLs that remain infected after a period of time. Because some websites remain infected at the end of the study, we cannot directly measure this probability but must estimate it instead. Differences between treatment groups were evaluated using the log-rank test. Additionally, a Cox proportional regression model was used to obtain the hazard ratios (HR). All two-sided  $p$  values less than 0.05 were considered significant.

### A. Measuring the impact of notices

First, we determined whether sending notices to hosting providers and domain owners had an impact on the cleanup of the infected URLs. Table IV provides some summary statistics regarding the status of the infected URLs 16 days after the notification. Entries are given for each treatment group. We reported the percentage of websites that were clean and the median number of days required to clean up those sites.

It is worth noting the significant difference between the two malware campaigns that took place during our experiment. From table IV, we can see that while 35% of the websites in the control group were clean after 16 days during the first campaign, only 26% of the websites in the control groups during the second campaign remediated their infection. The same trend was observed for the rest of the treatment groups, i.e., lower cleanup rates were achieved during the second campaign than during the first campaign. For instance, the percentage of remediated infections for the high-reputation group was reduced from 81% in the first campaign to 49%

Treatment type	Campaign 1			Campaign 2		
	#	% clean	Median clean up time	#	% clean	Median clean up time
Control	17	35.29%	14 days	229	26.20%	8 days
Indiv. researcher	23	69.57%	4 days	57	49.12%	2.5 days
University	17	64.71%	4 days	61	44.26%	3 days
Anti-malware Org.	20	80.95%	2 days	62	48.39%	1.5 days

TABLE IV: Summary statistics on the time to clean up, according to the treatment group

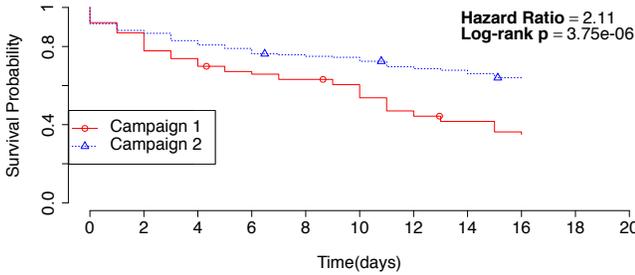


Fig. 4: Survival probabilities for each notification campaign. The overall cleanup rates are lower in the second campaign when infections were harder to verify by providers.

in the second campaign. We attribute these differences to the behavior change of the Asprox botnet which became harder to identify and remove during the second campaign (see Section III).

To further investigate whether these differences are significant, we compute the survival probabilities for each of the two different campaigns. Figure 4 plots these curves. This figure shows that 36% of websites that were notified during the first campaign remained infected after 16 days, compared to 65% for those that were notified during the second campaign. The log-rank test corroborated that the cleanup rate was significantly different during the two campaigns ( $\chi^2 = 21.39, p = 3.75e - 06$ ). Proportional hazard model was used to compute the adjusted-hazard ratio (HR) for the two campaigns with 95% confidence intervals (CI). The HR for remediating the infection in the first campaign was 2.11 (95%CI, 1.52-2.89) versus the second campaign, i.e., infected domains in the first campaign were cleaned up 2 times faster than during the second campaign. As both campaigns had significantly different cleanup rates, in the following we analyze them separately.

1) *Campaign 1*: Comparing the percentage of clean websites of the control group with the other treatment groups, we can estimate whether the notices made a difference in terms of expediting the cleanup. As shown in Table IV, the control group always achieved a lower percentage of clean websites than the other groups. For instance, the median number of days to clean an Asprox-infected website was 14 days when no notice was sent. However, the median number of days to remediate an infection was greatly reduced when notices were sent. Websites in the high-reputation group were cleaned after 4 days in average. This supports the hypothesis that notices expedite the cleanup process.

Group	Control		Indiv. researcher		University		Anti-malware Org.	
	$\chi^2$	p-value	$\chi^2$	p-value	$\chi^2$	p-value	$\chi^2$	p-value
Control			8.2	0.0041	6	0.0139	17.1	0.00003
Indiv. researcher	8.2	0.0041			0.2	0.644	1.7	0.198
University	6	0.0139	0.2	0.644			2.8	0.0972
Anti-malware Org.	17.1	0.00003	1.7	0.198	2.8	0.0972		

TABLE V: Log-rank test results (Campaign 1)

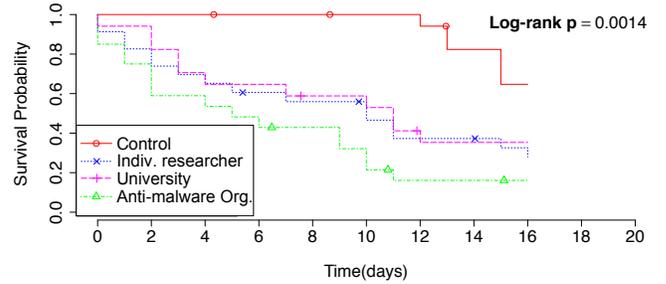


Fig. 5: Survival probabilities per treatment group (Campaign 1)

Again, to assess whether these difference are significant, we compute the survival probabilities for the different treatment groups (see Figure 5). We can observe different cleanup rates between the control group and the treatment groups which received notices. This figure shows that 65% of websites that were not notified remained infected after 16 days, compared to 30%, 35%, and 19% for those that belonged to the low-reputation, medium-reputation and high-reputation group respectively. The log-rank test confirms that these differences between the groups that received notices and the control group are significant ( $\chi^2 = 15.61, p = 0.0014$ ). However, the differences among any of treatment groups which received notifications are not significant (see Table V).

2) *Campaign 2*: In the previous section, we analyzed the impact of the notices that were sent during the first campaign and proved that sending notices expedited the cleanup process. In the following, we analyzed the impact of the notices sent during the second campaign that took place during January 2015.

As shown in Table IV, during this second campaign the percentage of sites successfully remediated was lower than during the first campaign. The control group had the lowest percentage of remediated infections, i.e., only 26% of websites were cleaned up. The rest of treatment groups achieved similar percentage of remediated sites (44%-49%). Therefore, though notices did impact the cleanup process, the reputation of the sender did not significantly affect that process.

Despite having a lower overall cleanup ratio, the sites that

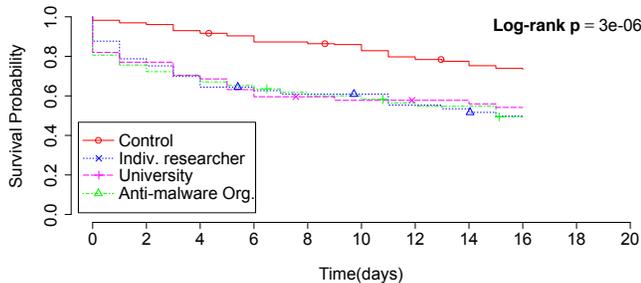


Fig. 6: Survival probabilities per treatment group (Campaign 2)

were remediated during the second campaign were cleaned up faster than in the first campaign. The median number of days before cleanup took place was 4 days during the second campaign, while it took 11 days during the first campaign. This suggests that the Asprox infections during the second campaign were harder to identify, but when detection was successful, clean up was done faster.

A plausible explanation for this pattern is to see it as the outcome of competency of the hosting provider. Those that are willing and able to recognize the compromise are also the ones that will be faster in terms of doing cleanup. Those that are not willing and able, will be slower in cleaning up or not do it at all. This explanation is consistent with the differences in cleanup between the two campaigns: at that time the malicious files of Asprox were easier to uncover, more hosting providers were able to initiate cleanup, including the less competent ones. The latter are likely to act more slowly, raising the median cleanup time.

We compute the survival curves for this second campaign per treatment group. Figure 6 plots the Kaplan-Meier estimates. In this campaign, the similarity among the treatment groups that received notices is even more clear than in the first campaign. This figure shows that after 5 days after tracking begun, 90% of websites that were not notified remained infected, compared to 64%, 63% and 65% for those that belonged to the low-reputation, medium-reputation and high-reputation group respectively. The log-rank test confirms that these differences between the treatment groups and the control group are significant ( $\chi^2 = 28.39, p = 3.01e - 06$ ). However, the differences among any of treatment groups are not significant (see Table VI).

Group	Control		Indiv. researcher		University		Anti-malware Org.	
	$\chi^2$	p-value	$\chi^2$	p-value	$\chi^2$	p-value	$\chi^2$	p-value
Control								
Indiv. researcher	17.1	3.51e-05	17.1	3.51e-05	13.6	22.1e-05	18.8	1.43e-05
University	13.6	22.1e-05	0.1	0.746	0.1	0.746	0	0.919
Anti-malware Org.	18.8	1.43e-05	0	0.91	0.2	0.678	0.2	0.678

TABLE VI: Log-rank test results (Campaign 2)

Therefore, though the notices were effective during both campaigns, the clean-up rates were higher during the first campaign. In neither of the campaigns did we observe a significant impact of sender reputation.

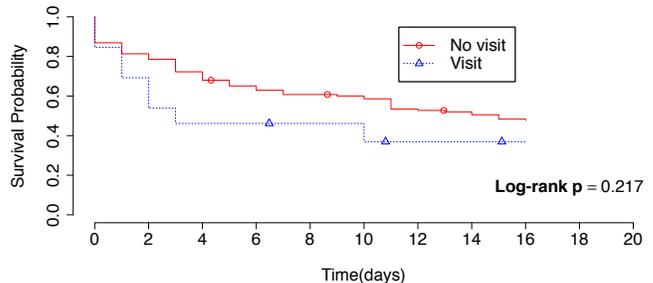


Fig. 7: Survival probabilities per cleanup website hosting provider visits

### B. Efficacy of the clean-up advice websites

As part of the experiment, we created three websites to assist the cleanup process. The corresponding link to these website was included in the abuse report. As it turns out, few recipients clicked the link. During the 16-day follow-up, we tracked the visitors to the web pages at the university and the free hosting site.<sup>1</sup> The number of visitors is presented in Table VII. As can be seen, only 8.97% of the hosting providers visited our cleanup website. Similarly, only 7.48% of the contacted website owners visited our cleanup website.

Treatment type	Campaign 1		Campaign 2	
	Host. Provider	Owner	Host. Provider	Owner
University	4	1	5	3
Indiv. researcher	1	2	3	5

TABLE VII: Number of cleanup website visitors per treatment group.

To analyze if of the cleanup websites did help expedite remediation, we measure the difference among visitors and non-visitors in terms of cleanup rates. The average cleanup time for the hosting providers that visited one of our websites was around 2 days, while for non-visitors it was almost 5 days on average. This decrease in average cleanup time may indicate a positive impact of the cleanup website. To further analyze the impact of this variable on the cleanup process, we estimate the survival probabilities for hosting providers that visited versus those who did not visited the cleanup website (see Figure 7). This figure shows that after 3 days, those hosting providers that visited one of the cleanup websites had already cleaned 53.8% of the infected domains, while those who did not visit any of our cleanup websites had only cleaned 28.8% of the infected websites after 3 days. However, though the cleanup rate is quite different during the first 3 days since the notice was sent, the survival curves are not significantly different (Log-rank test:  $\chi^2 = 1.5, p = 0.214$ ). Thus, after the 16-day followup the cleanup rate of the hosting providers that visited our websites is not significantly different from the cleanup rate of those who did not visit our website.

<sup>1</sup>We were unable to track the visitors of the StopBadware website due to Cloudflare cache management.

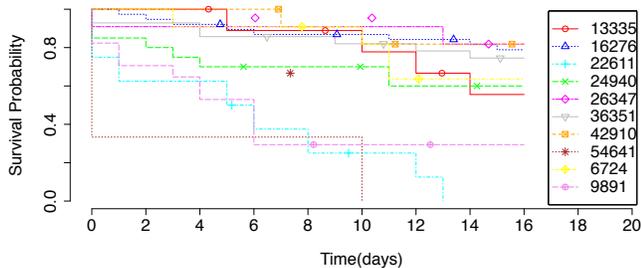


Fig. 8: Survival probabilities top 10 autonomous systems

This also suggests that hosting providers have different policies to deal with website infections. Table VIII describes some basic statistics of the top 10 autonomous systems in terms of number of Asprox infected domains. We can see clear differences both in terms of the amount of remediated infections and also in terms of average time to clean up an infected website. For instance, 'InMotion' hosting provider remediated all the infection in less than 4 days in average, while 'OVH' only remediated 21.05% of the websites and took around 8 days on average for those it did clean up. Figure 8 plots the survival curves for these hosting providers. Again, we can see significant different in terms of cleanup rate for the different hosting providers. 'InMotion', 'CS Loxinfo' and 'Hetzner' had cleaned more than 20% of their infected websites after 5 days while the rest of hosting providers took more than 10 days to achieve a similar percentage.

AS Name	#AS	# Infections		% clean		Avg. Cleanup Time (days)		CC
		Camp. 1	Camp. 2	Camp. 1	Camp. 2	Camp. 1	Camp. 2	
CloudFlare	13335	0	9	-	44%	-	10.25	US
OVH	16276	9	29	22.22%	21%	10.00	7.29	FR
InMotion-West	22611	2	6	100.00%	100%	7.00	5.17	US
Hetzner	24940	5	15	100.00%	20%	5.20	1.67	DE
Dreamhost	26347	0	6	-	33%	-	6.50	US
SoftLayer	36351	3	25	66.67%	20%	8.33	4.40	US
SadecceHosting	42910	2	9	50.00%	11%	10.00	7.00	TR
InMotion	54641	0	6	-	100%	-	3.33	US
Strato	6724	1	12	100.00%	25%	10.00	5.40	DE
CS Loxinfo PLC	9891	0	17	-	71%	-	3.08	TH

TABLE VIII: Summary cleanup statistics per AS owner.

Similarly, we measured whether website owners that visited our websites were capable of cleaning their infected websites faster. The average cleanup time for the website owners that visited one of our websites was 4.20 days in average, while for those who did not visit a cleanup website it was 4.26 days in average – an insignificant difference. The same result is shown by the survival probabilities (see Figure 9). After 7 days, the owners who visited the site had cleaned 36.4% of the infected domains, while those who did not visit cleaned 40.8% of the websites after 7 days. Thus, visiting the cleanup website did not make a difference for the website owners (Log-rank test:  $\chi^2 = 0.2, p = 0.648$ ). In short, it seems providing cleanup advice is not helpful, at least not in this form. If we assume that less technically competent owners are more likely to follow the link, then even basic advice does not enable them to achieve better cleanup.

These results suggest that: i) hosting providers play a major role when it comes to remediating an Asprox infection, ii)

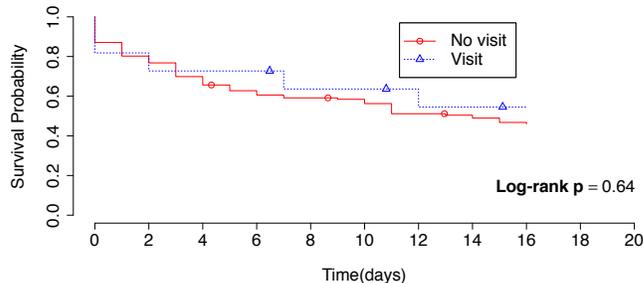


Fig. 9: Survival probabilities per cleanup website owner visitors

hosting providers that visited our cleanup website correlated to a higher rate of remediating the infection that those that did not, and iii) website owners seem to not have enough skills to clean up their own website once it gets infected, even when basic suggestions are provided.

### C. Analyzing responses from notified parties

During our experiment, we contacted 480 abuse contacts and received e-mail responses from 89 contacts. Of these 11 (12%) were clearly from a human, while 78 (88%) were machine-generated. The vast majority of responses were in English. Other common languages included Chinese, Russian, German, French, Turkish, Iranian, Thai, and Spanish.

Automated messages came in two forms: confirmations (28%) and tickets (72%). Confirmation e-mails simply acknowledge receiving our notification. Tickets provided a reference or ticket identifier associated with our notification message.

Throughout the experiment, 173 out of 240 notifications we sent to site owners bounced back mostly due to lack of abuse@domain address. On the other hand, the same addresses belonging to hosting providers bounced back once, indicating that the vast majority of hosting providers were at least setup to receive abuse e-mails. The difference can be explained in terms of awareness, technical knowledge, and/or liability. Whereas site owners are likely not aware of abuse reporting conventions, lack technical knowledge, and generally are not held liable for the distribution of malicious content, hosting providers as organizations generally are aware, and also potentially liable [14].

We investigated the relationship between the responses of notified parties and their cleanup behavior. Table IX provides some summary statistics regarding the status of the infected URLs after 16 days according to each response type that we received. Entries are given for each treatment group. Again, we reported the percentage of websites that have been found clean at the end of our 16-day investigation and the median number of days required to clean up those sites. We cannot observe any significant difference in the number of received responses across the treatment groups. This suggests that none of the notified entities decided whether to reply based on the reputation of the sender.

Treatment Group	Campaign 1						Campaign 2					
	Human responses			Automated responses			Human responses			Automated responses		
	#	% clean	Median Cleanup	#	% clean	Median Cleanup	#	% clean	Median Cleanup	#	% clean	Median Cleanup
Indiv. Researcher	3	100%	1 day	7	86%	5 days	1	100%	1 day	16	56%	13 days
University	1	100%	2 days	5	60%	12 days	4	75%	5 days	23	57%	4 days
Anti-malware Org.	1	100%	6 days	7	100%	2 days	1	100%	4 days	20	60%	4 days

TABLE IX: Summary statistics on the cleanup time according to the type of response

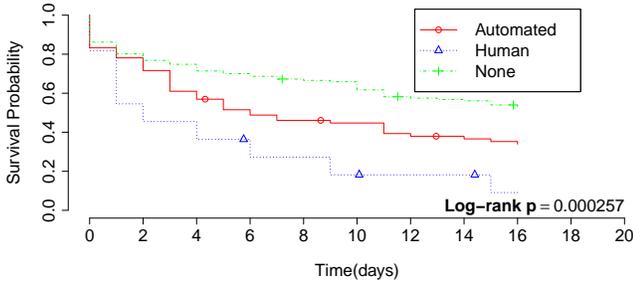


Fig. 10: Survival probabilities per response type

We did, however, find statistically significant differences between each of the type of responses and cleanup rates (Log-rank test:  $\chi^2 = 16.6, p = 0.000247$ ). As shown in Figure 10, within four days after notification, 64% of human responders had already cleaned up their websites, while automated responders had remediated 43% of the infections, and those parties that didn’t reply at all had only cleaned 29% of the compromised sites. Thus, the second strongest reactions came from contacts configured to send automated responses. This indicates that hosting providers using a system to automatically process notifications and complaints are more likely to act. As expected, the least effective reaction came from those hosting providers that never responded. After the first week, only 32% of such contacts had conducted some remediation; after 16 days, 48% had. While these cleanup rates are lower, they do show that even when hosting providers do not respond, it does not imply they ignored the message.

## V. RELATED WORK

A few researchers have recently begun investigating how notifications about system compromise or vulnerability can promote remediation. Most similar to our own work, Vasek & Moore conducted an experimental study on web-based malware URLs submitted to the StopBadware community feed [1]. They found that abuse reports sent with detailed information on the compromise are cleaned up better than those not receiving a notice (62% vs. 45% cleaned after 16 days). Moreover, they found no difference between the cleanup rates for websites receiving a minimal notice and those not receiving any notice at all. Based on this finding, we elected to provide detailed information in the abuse reports we sent. Thus, we corroborate their finding that detailed notices work on a different type of incident dataset.

Furthermore, we studied how different forms of notifications affected uptimes of malware cleanup rates [15]. To this end,

we compared the uptimes of ZeuS command and control servers provided by Zeus Tracker, Cybercrime Tracker and a private company. ZeuS Tracker and Cybercrime Tracker present a publicly accessible dynamic webpage that displays ZeuS malware command and control servers. On the other hand, the private company did not publicize any of detected command and control servers. We showed that publicized command and control servers were mitigated 2.8 times faster than the ones that were not publicized.

Another malware-orientated study supported the notion that notifications spur intermediaries to take action: in Canali et al. [2], researchers setup vulnerable webservers and compromised them. After a period of 25 days, they notified their own web hosts. Approximately 50% took action, generally suspending access. To ensure that the notifications were actually being read and not simply being acted upon without evidence, false abuse reports were also sent, resulting in 3 of the 22 providers suspending an account without actual evidence. This in turn suggests that most, but not all, recipients investigate abuse reports before taking action.

Whereas the present work and studies described above focus on reports of compromise, other researchers have sent notifications to the operators of vulnerable, but not necessarily compromised, systems. The goal here is to patch the vulnerable systems instead of remediating an infection. For example, Durumeric et al. notified hosts vulnerable to the widely reported Heartbleed vulnerability [3]. After scanning and excluding device and large-scale cloud providers (such as Amazon), researchers automatically identified 4,648 unique administrative contacts for 162,805 vulnerable hosts. They then divided the contacts into a treatment group receiving notifications and a control group that did not (at least initially). The treatment group was notified by e-mail and pointed to a detailed patching guide hosted at a University website. The researchers observed a 39.5% rate of patching for those receiving notifications, versus 26.8% for those that did not.

Similarly, Kührer et al. issued notifications for systems vulnerable to DDoS amplification attacks involving NTP [4]. Rather than directly notify each individual host with information about the vulnerability, the researchers provided lists of afflicted IP addresses to key organizations such as abuse team contacts at CERTs, security data clearinghouses such as Shadowserver, and afflicted vendors such as Cisco. They complemented this effort by working with CERTs to issue informative advisories warning of the vulnerability and how to patch affected systems. This multi-pronged approach proved very effective: they observed a 92% reduction in amplifiers

after three months tracking a population of 1.6 million affected hosts. Although the authors did not design an experiment with a control group, the researchers credited the campaign's success to collaboration with reputable sources who then issue notifications. This suggests that sender reputation might be influential after all, despite the negative findings from our study. In future work, we recommend investigating alternative sources of reputation, such as other intermediaries capable of coordinating cleanup and/or the use of private contact details for sharing compromise information.

Finally, with respect to general e-mail spam, a quasi-experiment by Tang et al. [16] saw researchers use two blocklists to compile a large source of e-mail spam and publish aggregated measures on SpamRankings.net. They then published the results for a treatment group and withheld results for a control group, observing a 15.9% reduction in spam among the treated group. Rather than notify individual hosts in order to remediate infections, the researchers' strategy relied on public shaming. The study indicates that abuse information could provide incentive for intermediaries to cooperate in remediating abuse on their networks.

## VI. LIMITATIONS

A number of limitations may impact the findings from our study.

First, we selected contacts to notify by inspecting the WHOIS for affected domains. Many abuse reports are sent between personal contacts, not general contact addresses, but we were unable to capture the impact of reputation in these trusted interactions. Our findings, therefore, apply only to the baseline case where personal contact has not been established. To put it differently, we are not claiming that reputation does not matter. Not only did an earlier study suggest it might (see section V), but the actual practices of abuse reporting show this every day. For example, many providers work with trusted reporters. In some cases, these notifications are trusted enough to allow for automated countermeasures or takedown actions.

Second, we measured reputation by the domain associated with the notification and the website used for cleanup advice. One potential issue is that our University-affiliated address was `tudelft.nl`, as opposed to the more widely known `.edu` top-level domain.<sup>2</sup> Nonetheless, anyone visiting the website for cleanup advice would clearly see the association with a University, while those visiting StopBadware's website would see that it was a non-profit cybersecurity organization. However, this is only one way to measure reputation. Reputation can also be established by sending credible notifications over a period of time. Because none of the organizations in our study regularly send notifications, we were unable to measure reputation in this fashion. However, it is something that we hope to do in future work, provided that we can partner with an organization that regularly sends abuse reports.

Third, we relied on a source of compromised URLs focused specifically on the malware delivery component of a

single, long-established botnet. We made this design decision intentionally, in order to control for the natural variation that exists between different types of abuse data. For example, a hosting provider might prioritize cleanup of command and control infrastructure over hacked websites that deliver malware. Furthermore, advanced persistent threats, banking trojans and phishing sites could attract more attention from hosting providers due to the financial implications and potential liability. The impact of sender reputation may differ in these scenarios, and so we defer such investigations to future work.

Fourth, there is a chance that latent characteristics appeared disproportionately in the treatment groups that influenced the overall outcome. For example, hosting provider size and type (shared vs. dedicated) may influence cleanup rates, but we were unable to verify that the distribution of these features is proportionate among treatment groups.

Fifth, we did not study re-infection of previously cleaned websites. Frequently, websites are recompromised when the hole that let the attacker in the first time is not plugged [17]. Because we were primarily interested in measuring the response to abuse reports, we elected to ignore subsequent reinfections.

Finally, there are a number of characteristics closely related to reputation that we did not examine. For example, none of our reports carried any suggestions that punitive action may result for ignoring the report. By contrast, notifications sent by Google (who controls search results) or ISPs and hosting providers (who control Internet access) might carry more weight due to the implication that there could be consequences for inaction. We defer investigating these effects to future work.

## VII. CONCLUSION

In this paper, we described an experiment to measure the differences in cleanup among notifications from senders with differing reputations. We find no evidence that reputation, as measured by the sender's type of organization, influences cleanup rates. However, we do find that detailed notices results in better cleanup overall. This confirms earlier findings carried out on websites distributing drive-by-downloads [1].

Furthermore, we find that publicizing and linking to a cleanup website containing specific instructions improves the cleanup rate when hosting providers view the instructions. However, this same positive impact is not shared by resource owners who served as point of contact for their domains. This suggests that differences in technical proficiency influence the success of a notification. Finally, throughout the trial, reports that elicited personal responses from the affected parties achieved higher cleanup rates. This suggests that personal interaction may contribute to better cleanup.

The role of the attacker in evading detection also plays a big role in how effective cleanup can be. We presented evidence that when compromise could be easily verified, cleanup rates were much higher than when the attackers took steps to hide

<sup>2</sup>Moreover, in certain cases, e-mails from `.nl` and `.org` addresses get caught in spam filters, whereas those from Gmail get through.

the compromise. We plan to study this effect in greater detail in future work.

Moving forward, we recommend three specific areas of study to further build on the work of this paper: first, the content of the notification and the presence of punitive measures; second, studying how cleanup websites are actually used by resource owners and intermediaries in order to craft a more effective message; and finally, sending notifications for other aspects of the cybercrime ecosystem, including command and control.

## VIII. ACKNOWLEDGMENTS

The authors would like to thank the anonymous contributor for generously sharing the Asprox data feed. Also, we thank StopBadware (Bryan Gulachenski and Marie Vasek) for their kind support and TU Delft's ICT staff for their help. This publication was supported by a subcontract from Rutgers University, DIMACS, under Award No. 2009-ST-061-CCI002-06 from the U.S. Department of Homeland Security and by a grant from the Netherlands Organisation for Scientific Research (NWO), under project number 628.001.022.

## REFERENCES

- [1] M. Vasek and T. Moore, "Do malware reports expedite cleanup? An experimental study," in *Proceedings of the 5th USENIX Workshop on Cyber Security Experimentation and Test (CSET)*, 2012.
- [2] D. Canali, D. Balzarotti, and A. Francillon, "The role of web hosting providers in detecting compromised websites," in *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2013, pp. 177–188.
- [3] Z. Durumeric, J. Kasten, D. Adrian, J. A. Halderman, M. Bailey, F. Li, N. Weaver, J. Amann, J. Beekman, M. Payer *et al.*, "The matter of heartbleed," in *Proceedings of the 2014 Conference on Internet Measurement Conference*. ACM, 2014, pp. 475–488.
- [4] M. Kühner, T. Hupperich, C. Rossow, and T. Holz, "Exit from hell? reducing the impact of amplification DDoS attacks," in *USENIX Security Symposium*, 2014.
- [5] D. Crocker, "Mailbox Names for Common Services, Roles and Functions," RFC 2142 (Proposed Standard), Internet Engineering Task Force, May 1997. [Online]. Available: <http://www.ietf.org/rfc/rfc2142.txt>
- [6] "StopBadware," <http://www.stopbadware.org/>.
- [7] "The moz blog study: How searchers perceive country code top-level domains," <http://moz.com/blog/cc-tld-domain-study>.
- [8] "Best Practices for Reporting Badware URLs," 2011, <https://www.stopbadware.org/files/best-practices-for-reporting-badware-urls.pdf>.
- [9] J. d. T. Nart Villeneuve and D. Sancho, "Asprox Reborn," Trend Micro Incorporated, Tech. Rep., 2009.
- [10] "Searching with virustotal," <https://www.virustotal.com/en/documentation/searching/#getting-url-scans>.
- [11] "Sucuri malware labs - php error: Fatal error," <http://labs.sucuri.net/db/malware/php-error-fatal-error?v6>.
- [12] "Server header checker – SEO tools," <http://tools.seobook.com/server-header-checker>.
- [13] "Cloudflare content delivery network," <https://www.cloudflare.com/features-cdn>.
- [14] "National institute of standards and technology - stopbadware commentary on liability of web hosts for malware distribution," [http://www.nist.gov/itl/upload/StopBadware\\_Web-Hosting-Provider-Liability-for-Malicious-Content.pdf](http://www.nist.gov/itl/upload/StopBadware_Web-Hosting-Provider-Liability-for-Malicious-Content.pdf).
- [15] C. Gañán, O. Cetin, and M. van Eeten, "An empirical analysis of zeus c&c lifetime," in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*. ACM, 2015, pp. 97–108.
- [16] Q. Tang, "Improving internet security through social information and social comparison: A field quasi-experiment."
- [17] T. Moore and R. Clayton, "Evil searching: Compromise and recompromise of internet hosts for phishing," in *Financial Cryptography and Data Security*, ser. Lecture Notes in Computer Science, R. Dingledine and P. Golle, Eds., vol. 5628. Springer, 2009, pp. 256–272.

APPENDIX  
CONTENT OF ABUSE REPORTS AND CLEANUP WEBSITES

hxxp://poorcompromisedwebsite .com/user.php is currently being abused to spread malware. This means it may be placing Internet users at risk. Please investigate and take appropriate action to resolve or mitigate the threat.

Description: Asprox botnet dropper

Date/time of detection: 2014-12-07 at 00.31 (GMT+1)  
IP address at time of detection: 195.158.28.146  
Additional parties notified: abuse@poorcompromisedwebsite.com (site owner)

You are receiving this report because this was listed as the technical contact e-mail in the WHOIS record for 195.158.28.146. If you believe you have received this report in error, or for more information, please contact us at this address: abuse-reporter@stopbadware.org

Caution: Opening malware URLs in your browser can infect your computer. For security reasons, URLs in this e-mail have been modified by replacing http with hxxp and by adding a space before the firstdot(.)

=====  
ADDITIONAL INFORMATION  
=====

Detailed malware description:

URL accessed: hxxp://poorcompromisedwebsite .com/user.php?c=RwFGxB7fBPAjwwWISCS7T09bzqUT3  
Behaviour: Delivers malicious executables and ZIP files.  
Special condition: Only delivers malicious executables when accessed through Windows Internet Explorer.

Tips for cleaning & securing a compromised website:  
<https://www.stopbadware.org/asprox-cleanup-advice#7NSVRLZ>

Fig. 11: Example of anti-malware organization e-mail notification

hxxp://poorcompromisedwebsite .com/user.php is currently being abused to spread malware. This means it may be placing Internet users at risk. Please investigate and take appropriate action to resolve or mitigate the threat.

Description: Asprox botnet dropper

Date/time of detection: 2014-12-07 at 00.31 (GMT+1)  
IP address at time of detection: 10.1.5.3  
Additional parties notified: abuse@poorcompromisedwebsite.com (site owner)

You are receiving this report because this was listed as the technical contact e-mail in the WHOIS record for 10.1.5.3. If you believe you have received this report in error, or for more information, please contact us at this address: malwarereporter-tbm@tudelft.nl.

Caution: Opening malware URLs in your browser can infect your computer. For security reasons, URLs in this e-mail have been modified by replacing http with hxxp and by adding a space before the firstdot(.)

=====  
ADDITIONAL INFORMATION  
=====

Detailed malware description:

URL accessed: hxxp://poorcompromisedwebsite .com/user.php?c=OG30hQ5HtuQGGQ38fe744itfo/kMWBKwc+Wjn7UH5mo  
Behaviour: Delivers malicious executables and ZIP files.  
Special condition: Only delivers malicious executables when accessed through Windows Internet Explorer.

Tips for cleaning & securing a compromised website:  
<http://www.cleanup-advice.tudelft.nl/#WJUB5TG>

Fig. 12: Example of University e-mail notification

hxxp://poorcompromisedwebsite .com/error.php is currently being abused to spread malware. This means it may be placing Internet users at risk. Please investigate and take appropriate action to resolve or mitigate the threat.

Description: Asprox botnet dropper

Date/time of detection: 2014-12-07 at 00.31 (GMT+1)

IP address at time of detection: 112.78.8.33

Additional parties notified: abuse@poorcompromisedwebsite.com (site owner)

You are receiving this report because this was listed as the technical contact e-mail in the WHOIS record for 112.78.8.33. If you believe you have received this report in error, or for more information, please contact us at this address: malwarereporting@gmail.com.

Caution: Opening malware URLs in your browser can infect your computer. For security reasons, URLs in this e-mail have been modified by replacing http with hxxp and by adding a space before the firstdot(.)

=====

ADDITIONAL INFORMATION

=====

Detailed malware description:

URL accessed: hxxp://poorcompromisedwebsite .com/error.php?c=WhfXoeHz6uhPe0IqdCHdcaG2Fi/2U1Y/xYy11GMOm2Y

Behaviour: Delivers malicious executables and ZIP files.

Special condition: Only delivers malicious executables when accessed through Windows Internet Explorer.

Tips for cleaning & securing a compromised website:

<http://cleanup-advice.besaba.com/#MNVTUUT>

Fig. 13: Example of individual researcher e-mail notification

About Webmaster Help Data Get Involved Publications **Donate**



A nonprofit that makes the Web safer by fighting badware

**Report Badware**

Blog Forum

**HELP! MY SITE IS INFECTED.**

1,638,326 URLs currently blacklisted by our data providers

169,823 Sites we've helped de-blacklist

BADWARE SEARCH

### Asprox Cleanup Advice

This is a guide on how to identify and remove the malware toolkit called **Asprox** from your compromised website. You have been directed to this page because we detected that your website has been compromised with Asprox malware.

You have two basic options: clean-up your server yourself, or contact your hosting provider or another specialist for help. Below we outline the basic steps if you want to undertake clean-up yourself. Note that this guidance only covers the most common cases. Some cases may require further help from a security professional and/or your hosting provider. We recommend backing up your files before taking any additional steps.

After performing the clean-up, we strongly recommend you to adopt certain precautionary measures, to protect your site from being compromised again. These precautionary measures are listed in step 5.

Step 1. Change administrator passwords

Asprox bots execute SQL injection attacks to steal administrator passwords. You should always change passwords after a compromise as a precaution.

Step 2. Remove Malicious PHP Code

Step 2a. Remove the botnet dropper PHP script. The name of script can be derived from the URL included in the notification email that you have received. We show an example below. Note that in your case, the PHP script file name might be different from the one shown in the example.

Poorcompromisedwebsite.com/ staart.php? u=0HTc3pive7IG9T9n3IVsJA

Compromised Domain Botnet Dropper PHP Script Extension

Remove It

Step 2b. Identify hidden PHP scripts and code which can be used to redirect or insert malicious links into the pages of your site. Remove any kind of malicious code/file you find in your site. Hidden PHP code might not have a .php extension. Criminals can create malicious files and give them common names, such as query.js or jquery.js. To determine whether a given suspected php or .js file is malicious, check to see if it includes obfuscated code, such as code beginning with eval(gainstate(base64\_decode("...")); Such tricks are commonly employed. Also criminals can create a number of sub-folders on the site with names such as /logs/ and /temp/ and create malicious files in these folders. Sometimes malicious files have names like 'main' or 'deb97b89098277d63c041efb6be44' with no file extension to hide the purpose of the file and make them look like system files.

(a) Anti-malware Organization

Student portal | Employee portal | Contact

search

Departments and sections

Study Research Cooperation Current

### How to Remove Asprox Malware from Your Website

This is a guide on how to identify and remove the malware toolkit called **Asprox** from your compromised website. You have been directed to this page because we detected that your website has been compromised with Asprox malware.

You have two basic options: clean-up your server yourself, or contact your hosting provider or another specialist for help. Below we outline the basic steps if you want to undertake clean-up yourself. Note that this guidance only covers the most common cases. Some cases may require further help from a security professional and/or your hosting provider. We recommend backing up your files before taking any additional steps.

After performing the clean-up, we strongly recommend you to adopt certain precautionary measures, to protect your site from being compromised again. These precautionary measures are listed in step 5.

Step 1. Change administrator passwords

Asprox bots execute SQL injection attacks to steal administrator passwords. You should always change passwords after a compromise as a precaution.

Step 2. Remove Malicious PHP Code

Step 2a. Remove the botnet dropper PHP script. The name of script can be derived from the URL included in the notification email that you have received. We show an example below. Note that in your case, the PHP script file name might be different from the one shown in the example.

Poorcompromisedwebsite.com/ staart.php? u=0HTc3pive7IG9T9n3IVsJA

Compromised Domain Botnet Dropper PHP Script Extension

Remove It

Step 2b. Identify hidden PHP scripts and code which can be used to redirect or insert malicious links into the pages of your site. Remove any kind of malicious code/file you find in your site. Hidden PHP code might not have a .php extension. Criminals can create malicious files and give them common names, such as query.js or jquery.js. To determine whether a given suspected php or .js file is malicious, check to see if it includes obfuscated code, such as code beginning with eval(gainstate(base64\_decode("...")); Such tricks are commonly employed. Also criminals can create a number of sub-folders on the site with names such as /logs/ and /temp/ and create malicious files in these folders. Sometimes malicious files have names like 'main' or 'deb97b89098277d63c041efb6be44' with no file extension to hide the purpose of the file and make them look like system files.

(b) University

HOME **HOW TO REMOVE ASPROX MALWARE FROM YOUR WEBSITE** EXTERNAL CLEAN UP AND UPDATE SITES CONTACTS

## How to Remove Asprox Malware from Your Website

This is a guide on how to identify and remove the malware toolkit called **Asprox** from your compromised website. You have been directed to this page because we detected that your website has been compromised with Asprox malware.

You have two basic options: clean-up your server yourself, or contact your hosting provider or another specialist for help. Below we outline the basic steps if you want to undertake clean-up yourself. Note that this guidance only covers the most common cases. Some cases may require further help from a security professional and/or your hosting provider. We recommend backing up your files before taking any additional steps.

After performing the clean-up, we strongly recommend you to adopt certain precautionary measures, to protect your site from being compromised again. These precautionary measures are listed in step 5.

Step 1. Change administrator passwords

Asprox bots execute SQL injection attacks to steal administrator passwords. You should always change passwords after a compromise as a precaution.

Step 2. Remove Malicious PHP Code

Step 2a. Remove the botnet dropper PHP script. The name of script can be derived from the URL included in the notification email that you have received. We show an example below. Note that in your case, the PHP script file name might be different from the one shown in the example.

Poorcompromisedwebsite.com/ staart.php? u=0HTc3pive7IG9T9n3IVsJA

Compromised Domain Botnet Dropper PHP Script Extension

Remove It

Step 2b. Identify hidden PHP scripts and code which can be used to redirect or insert malicious links into the pages of your site. Remove any kind of malicious code/file you find in your site. Hidden PHP code might not have a .php extension. Criminals can create malicious files and give them common names, such as query.js or jquery.js.

To determine whether a given suspected php or .js file is malicious, check to see if it includes obfuscated code, such as code beginning with eval(gainstate(base64\_decode("...")); Such tricks are commonly employed. Also criminals can create a number of sub-folders on the site with names such as /logs/ and /temp/ and create malicious files in these folders. Sometimes malicious files have names like 'main' or 'deb97b89098277d63c041efb6be44' with no file extension to hide the purpose of the file and make them look like system files.

More examples of malicious PHP code can be seen here: <http://av-snap.info/articles/php-examples.php>.

Step 3. Remove hidden HTML Elements

Placing hidden malicious links on the pages of websites is a common tactic among cyber-criminals. Hackers will place the links in a html element that can be "hidden" using CSS such as a <div> <span> <iframe> or even a list <ul>. This type of hack is fairly easy to spot when viewing the source code of the page (type Ctrl+U in most browsers). To clean up this hack, simply delete the malicious links from the pages on the website. Commonly, these hidden elements can be found at the very beginning of the file, before the document type declaration or <html> tag and/or after the closing </html> tag

(c) Individual researcher

Fig. 14: Cleanup websites