Do malware reports expedite cleanup? An experimental study

Marie Vasek
Computer Science & Engineering Dept.
Southern Methodist University, Dallas, TX
marie.vasek@gmail.com

Tyler Moore
Computer Science & Engineering Dept.
Southern Methodist University, Dallas, TX
tylerm@smu.edu

Abstract

Web-based malware is pervasive. Miscreants compromise insecure hosts or even set up dedicated servers to distribute malware to unsuspecting users. This scourge is mainly fought by the voluntary action of private actors who detect and report infections to affected site owners, hosting providers and registrars. In this paper we describe an experiment to assess whether sending reports to affected parties makes a measurable difference in cleaning up malware. Using community reports of malware submitted to StopBadware over two months in Fall 2011, we find evidence that detailed notices are immediately effective: 32% of malware-distributing websites are cleaned within one day of sending a notice, compared to just 13% of sites not receiving a notice. The improved cleanup rate holds for longer periods, too – 62% of websites receiving a detailed notice were cleaned up after 16 days, compared to 45% of websites not receiving a notice. It turns out that including details describing the compromise is essential for the notice to work – sending reports with minimal descriptions of the malware was found to be roughly as effective as not sending reports at all. Furthermore, we present evidence that sending multiple notices from two sources is not helpful. Instead, only the first transmitted notice makes a difference.

1 Introduction

The web is a leading vector for infecting computers with malware. We distinguish between three main approaches to distributing malware on the web: compromised websites, free web hosting, and purely malicious websites. The most common approach is for a miscreant to compromise a legitimate website using a variety of techniques, such as by exploiting code injection vulnerabilities or leveraging stolen FTP credentials. One key advantage of compromising legitimate websites for attackers is that regular visitors to the website can be exposed during the course of routine browsing. Some attackers do not even bother compromising websites. Instead, they sign up sham accounts and place malware on free-hosting websites, which offer web space to all comers.

The final distribution option is for the miscreant to configure a website directly under their control. Such purely malicious websites exist only to deliver malware. Often, compromised websites are used in combination with purely malicious websites, where the compromised website will automatically redirect to a malicious site or remotely load the malicious payload hosted on the purely malicious website [6].

A constant battle is engaged between malware peddlers and Internet operators over the infection and remediation of websites. One striking feature of this back and forth is the largely voluntary nature of exchanging information on incidents and cleaning up infected websites. We became interested because there have been community efforts to devise standards for reporting [9]. These efforts recommend increased levels of detail to be provided to operators whose websites have been infected. Implicit in these recommendations are two key assumptions. First, the recommendations assume that website operators will usually cooperate in the remediation if provided information on infections. Second, more detailed information is better, as it lends added credibility to the report and therefore makes action more likely.

We set out to test these assumptions experimentally. In particular, we hypothesize that transmitting detailed reports of malware to affected parties will cause the reported websites to be remediated more quickly and comprehensively than if no report were sent.

Confirming or refuting this hypothesis is important, given the limited resources available to website operators and those investigating malware incidents. Sending out detailed reports can be quite resource-intensive, particularly since it requires investigating compromises that can vary considerably in method and involve a sequence of affected websites. Furthermore, ascertaining which par-

ties should receive the reports and tracking down their contact details can be time consuming. Consequently, it is important to provide reliable evidence of the effectiveness of notices. If the notices are shown to lead to better remediation outcomes, then the expense of disseminating notices may be justified.

Section 2 explains how designed and carried out the experiment using malware reports submitted to StopBadware over two months in Fall 2011. In Section 3 we present the results. We find that detailed reports do in fact expedite malware cleanup substantially. Within one day of receiving a detailed report, 32% of malware-distributing websites have been cleaned, compared to just 13% of websites that did not receive a notice. We also find that the additional detail of incidents is essential, as less-detailed reports appeared about as effective as not sending any reports at all. We review related work in Section 4 before drawing conclusions and outlining opportunities for further investigation in Section 5.

2 Experimental methodology

In order to determine whether sending malware notices reduces the time it takes to clean up affected sites, we carried out a three-phased procedure illustrated by the flow charts in Figure 1. In the first phase (a), we process incoming reports to identify sites with malware. In the second phase (b), contact information for affected parties is found and the sites are randomly assigned to the control, minimal and full notification groups. In the third phase (c), sites are periodically reevaluated to determine if and when malware has been removed.

2.1 Processing incoming malware reports

The procedure for processing incoming reports follows the flow chart in Figure 1 (a). We gathered reports of malware URLs from the `BadwareBusters.org` community feed[1], where any Internet user can report a malware URL to StopBadware. We then pared down the reports by removing duplicates. We also exclude reports from `*.c[a-z].cc`, `*.ce.ms`, and anything on the `dropbox.com` domain, since the operators of these sites already have established prior agreements to act on notices from StopBadware.

We next determine if the URL is actively delivering malware. This process is discussed in Section 2.4. We then classify every unique instance which helps deliver malware as a candidate for reporting. For our purposes, we define a unique instance as coming from a single IP address, second-level domain name, or subdomain of a known free web hosting service. For example, if `bad.example.com` redirected using HTTP status code 301 to `bad2.example.com`, and both URLs had the same IP address, we would count them as a single malware instance. The only exception would be if `example.com` was a free site hosting domain, in which case they would be considered unique.

We also frequently identify compromised hosts that load malicious code hosted on external websites. In these cases, we add both sites to the candidate list.

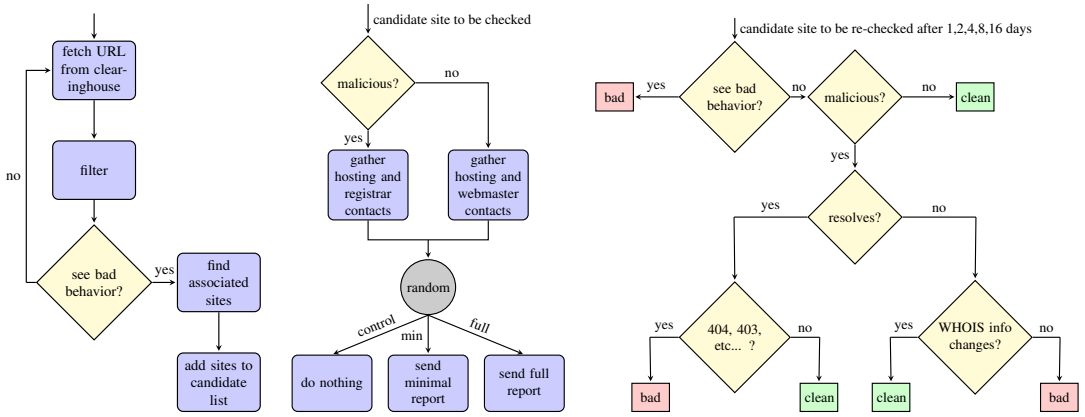
2.2 Notifying affected parties

The procedure for notifying affected parties follows the flow chart in Figure 1 (b). We first determine whether a candidate site has been compromised or set up purely for malicious purposes, as this affects which parties are sent notices. We manually visit the associated second-level domain (e.g., if `my.example.com/ajaxam.js` delivers malware, we would examine `example.com` in a browser). If the top-level website appears legitimate, we would deem the site compromised. However, if the top-level site looks dubious, such as returning a default Apache server page, or a 200 page with no content, then we deem the site to be registered for primarily malicious purposes. If the top-level site otherwise appears not intended for end users to visit (e.g., delivering a 404 error or a default WordPress install), then we run the site through `google.com`. If search results return pages of what appears to be legitimate content, we consider the site to be infected rather than purely malicious. Otherwise, particularly if Google turns up only malware complaints, we consider the site to be malicious. We also examine WHOIS information for the domain at this stage to confirm information that can be found through Google. If the WHOIS information is bogus (the address is not a real place or otherwise appears obviously fake), we deem the site malicious.

We next decide which operators should receive notices. If the malware is hosted on a compromised website or free-hosting service, we send notices to the site owner and hosting provider. To look up contact information for the site owner, we first check the website itself for an email address. If we cannot find one, we next look up the WHOIS on the domain. If that does not contain an email address, then we use the default email addresses for a domain (`abuse@example.com` and `webmaster@example.com`).

If the website is deemed to be purely malicious, we send notices to the hosting provider and domain name registrar. To contact the domain registrar, we look up WHOIS information on the reported domain name. If the registrar includes a contact email address there, then we use it. Failing that, we look at the registrar's website and do a Google search for "{domain registrar} abuse con-

[1] `https://badwarebusters.org/community/submit`



(a) Processing malware reports (b) Notifying affected parties (c) Following up to determine when clean

Figure 1: Flow charts describing the experimental design.

tact." When all else fails, we use the contact information given by IANA that all registrars must provide.

For all sites, we look up contact information for the hosting provider by collecting WHOIS information on the IP address. In our study, all sites included the requisite information in the WHOIS report.

Finally, we randomly assigned each site to one of three groups: control, minimal, and full. Sites assigned to the control group receive no report. Sites assigned to the minimal group are sent an email report which contains the minimum amount of information required to satisfy the best practices [9] (URL, short description of malware, IP address, date and time malware detected, email address of reporter) along with why that person was the target of a malware report. Sites assigned to the full group receive all the information from the minimal report, plus a longer and more detailed description of the malware along with any special information needed for the malware to deliver (e.g., a specific HTTP referrer).

2.3 Following up to estimate cleanup time

We regularly reinspect sites distributing malware to measure the time it takes to clean sites.Because the inspection is performed manually, we only carry it out at fixed intervals of 1, 2, 4, 8 and 16 days following the initial report. We continue to follow up even after a site has been found to be clean, since sites are often immediately reinfected. We only consider a site clean once it is never subsequently found to be bad.

We begin the followup procedure outlined in Figure 1 (c) by checking to see if the site is still bad. If the site appears clean and has not been classified as purely malicious, then we mark the site clean and conclude our assessment. However, if the site purely exists for malicious purposes and we cannot find malware, we must continue our investigation since many such sites attempt to hide malicious behavior from investigators. If the site does not resolve, then we check the WHOIS record to see if the responsible parties took the site down. If there is no change, then we assume that the malicious actors are still present and consider the site bad. If the site resolves but delivers suspicious behavior such as a 404 or 403 page, then we consider the site bad. However, if the site is delivering a hosting provider interstitial page or something similar, we deem the site to be clean.

2.4 Assessing for malware

In the previous subsections, we explained that we check whether a site is *bad*. We now describe in detail how we make this determination. The basic procedure is given by the flow chart in Figure 2.

We start by opening the site in a browser in a virtual machine (VM) and capture all HTTP traffic using a packet logger. We access the URL with a `google.com` HTTP referrer, or other referrer depending on the context given in the initial clearinghouse report.

If the reported site appears to deliver a malicious executable, we download the executable and run it through VirusTotal[2], an online service that compares executables against all major antivirus products. If at least 3 different vendors flag an executable as malicious, we conclude that the website is distributing malware.

We analyze the network traffic for anything out of the ordinary (e.g., draws code from a known malicious site or from a suspicious site). We also inspect the HTML and associated JavaScript and CSS files from the page for malicious injected code. Some malicious code renders in one browser but not another, so we do not rely only on observing attempted exploitation on our VM's browser.

[2] `https://www.virustotal.com/`

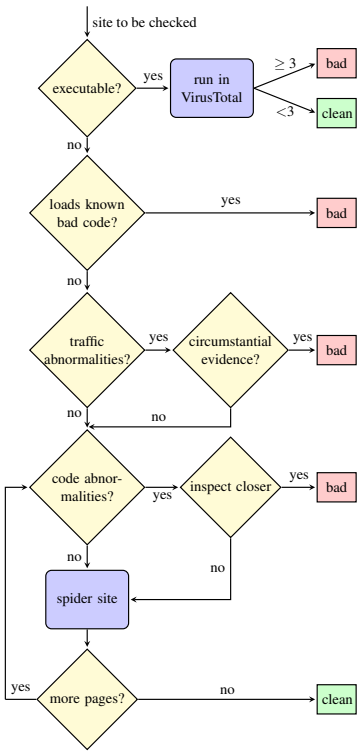


Figure 2: Flow chart for deciding whether a site is *bad*, i.e., distributing malware.

Finally, if we cannot find malware on the reported URL, we spider the site and manually assess the code on a representative sample of the URLs on the site. If this is an initial report, we report the first URL we find malware on along with up to two other URLs on the same site with malware. For follow-up investigations, we note the other URLs that deliver malware to check in any subsequent followups and consider this site still bad. Only after a site clears all these checks do we deem it clean.

3 Results

We examined reports submitted to the StopBadware Clearinghouse between October 10 and December 5, 2011. Of the 960 distinct reports submitted during that time, we identified 161 distinct instances of malware eligible for reporting.

3.1 Measuring the impact of notices

We first set out to determine whether sending notices affected the rates of malware cleanup. Table 1 provides some summary statistics to help answer this question. Entries are given for each category of notice – control (no notice issued), minimal, or detailed notices. The left-most figures summarize all 161 reports. We report the number matching each category, followed by the percentage of websites that have been found clean at the end of our 16-day investigation. Lastly, we report the median number of days required to clean up those sites that were successfully remediated.

For instance, we can see that 45% of websites in the control group are cleaned within 16 days, compared to 49% of those receiving a minimal notice and 62% of those receiving a detailed notice. This supports the hypothesis that notices do make remediation more likely. The data also suggests that the remediation, when it takes place, occurs faster when detailed notices are sent. Sites receiving detailed notices are cleaned up within one day (that is, if they are among the 62% of sites that are cleaned up at all), compared to two days for detailed notices and four days for websites in the control group.

The other figures in the table examine whether differences in the type of malicious behavior affect the notices. The table compares those websites deemed to be "purely malicious" to those websites that are merely compromised. In both cases, notices substantially improve cleanup rates – rising from 46% to 58% on purely malicious sites and 45% to 63% on compromised sites. Similarly, the distinction between sites directly distributing executables and others does not seem to substantially impact the effectiveness of notices. The figures tell a similar tale: more detailed notices lead to better cleanup within a shorter amount of time.

Because we periodically check whether an infected website has been remediated, we cannot know precisely when the cleanup occurs. Instead, the data on cleanup times is *interval-censored*. For example, if a website is still infected when we check after 1,2, and 4 days, but on the eighth day is found to be clean, we record the cleanup time as having occurred during the interval $(4,8]$. Notably, many websites remain infected even after 16 days. In this case, we record the cleanup time as having occurred between $(16,\infty)$.

The presence of interval-censored data means that we cannot directly compute statistics such as cumulative distribution functions. Instead, we must use survival analysis to estimate the expected probabilities based upon the recorded intervals. Figure 3 (left) plots a survival function for the three categories of notices. A survival function $S(t)$ measures the probability that an infection takes more than t days to clean. This is similar to a complementary cumulative distribution function, except that the time intervals are taken into account using a Turnbull estimator to compute the the probabilities [10].

The black dashed lines in the plot includes 95% confidence intervals for the control case based on the cumulative hazard function. The red dash-dotted line indicates the survival function for those sites receiving minimal

Report type	All badware			Purely malicious			Compromised			Executable			Other malware		
	#	% clean	days	#	% clean	days	#	% clean	days	#	% clean	days	#	% clean	days
Control	53	45	4	13	46	4	40	45	4	12	67	2.5	41	39	4
Minimal	55	49	2	17	53	4	38	47	1	6	50	8	49	48	1.5
Full	53	62	1	17	58	1.5	36	63	1	5	80	8.5	48	60	1

Table 1: Summary statistics on the time to clean up malware, according to the type of report issued. The table first presents results for all sites, then divided according to whether the site was deemed to be purely malicious and whether the website directly hosted a malicious executable file.

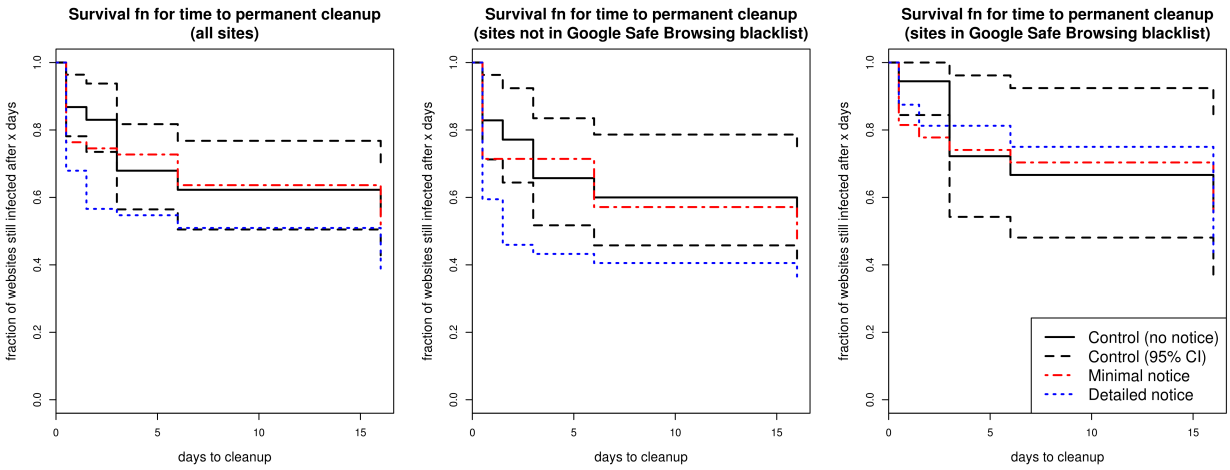


Figure 3: Survival function for cleanup time of infected websites: all sites (left), sites not in Google's blacklist (center), sites also in Google's blacklist (right).

notices, while the blue dotted line indicates the survival function for sites receiving full notices. 87% of websites that received no notice remained infected after one day, compared to 76% for those receiving minimal notices and 68% of those receiving full notices.

As can be seen from the graph, sites receiving full notices tend to be cleaned up sooner, and the reduction is nearly always greater than the 95% confidence interval for the control notices at each point. By contrast, the survival function for the minimal notices is very similar to the control notices, except after the first day. These results suggest that detailed notices expedite cleanup more than sending sparse notices or not sending notices at all. In the following subsections we examine additional factors that may affect the effectiveness of notices.

3.2 Accounting for outside notifications

One complicating factor when running an experiment measuring the impact of sending notices is that we are not necessarily the only party sending them. Any unobserved notices will necessarily weaken our findings, since for example if a website assigned to our control group receives a notice from someone else, then we are measuring the effect of that notice but assigning it to a group that was not supposed to receive a notice.

There are a few ways we can deal with this issue. First, we can observe that so long as the unobserved notices are relatively evenly distributed amongst our groups, then the effect will be felt across all groups and so the only impact of the unobserved notices would be to understate the effectiveness of notices.

Second, we can directly examine the impact of outside notifications on our results. To do that, we checked all websites against Google's Safe Browsing malware blacklist [1]. Google's malware blacklist is noteworthy because it is very extensive and since Google automatically notifies webmasters whenever their websites are found to be infected. Unsurprisingly, a malware notice from Google can quickly attract the attention of a webmaster dependent on traffic referrals from search engines. Consequently, we compare the effects of notice on reports that did not receive Google reports to those receiving Google reports.

Report type	**Not in Google** #	% clean	days	**In Google's BL** #	% clean	days
Control	35	46	3	18	44	4
Minimal	28	54	1	27	44	3
Full	37	65	1	16	56	16

Table 2: Summary statistics on cleanup rates based on whether a site appears in Google's blacklist. Sites not in Google's blacklist see the biggest gains from notices.

Table 2 shows that 100 of the 161 websites appearing in our data set did not appear in Google's blacklist. It also shows a relatively even distribution among the three categories of notices.

Perhaps the most striking result can be seen by inspecting Figure 3 (center), which shows the survival function for the websites that did not receive a notice from Google. In this case, the effect of sending full notices increases substantially. Now, the survival function for sites receiving full notices clearly exceeds the 95% confidence interval for the control group. The cleanup rate for full notices is consistently around 20 percentage points higher than for the control case. For instance, 17% of websites that did not receive a notice from Google or us were cleaned up within one day, compared to 40% of sites receiving a notice from us. Even after 16 days, when 46% of sites receiving no notice are cleaned up, 65% of websites receiving a detailed notice were cleaned up.

Finally, Figure 3 (right) shows the survival function for the websites that *did* receive a notice from Google. Here, our sending additional notices did not have any additional positive effect. Rather, the cleanup rate is virtually indistinguishable between sites receiving a notice and those that did not. This is somewhat surprising – one might expect that a website receiving multiple notices of infection from different sources would be more likely to take action. Instead, it appears that diligent website operators act on the first notice, while irresponsible ones ignore all notifications.

3.3 Does the nature of compromise affect cleanup?

We also wondered whether the nature of compromise might affect the cleanup rate. Websites that have been compromised often have a strong incentive to remove malware, particularly if search engines stop referring web traffic to their sites as a result. By contrast, websites that have been registered for malicious purposes will by definition be unresponsive, leaving the cleanup task to the domain name registrar or the hosting provider, which may not be keen to offend a paying customer. To that

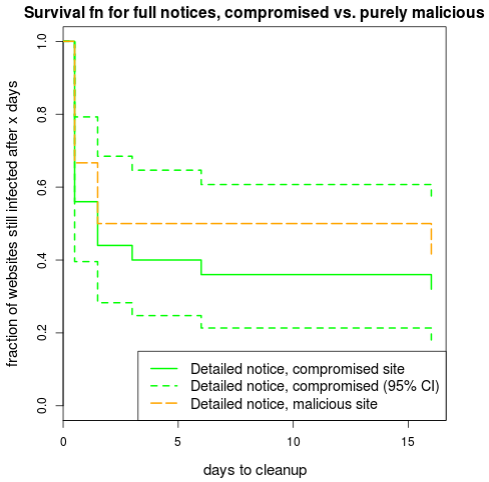


Figure 4: Survival function for cleanup time of compromised and purely malicious websites.

end, we now examine the evidence to answer the following two questions:

1. Do compromised hosts get cleaned up faster than purely malicious websites?

2. Does hosting the payload make a difference?

Recall that we distinguish between websites that are compromised from those that are registered for purely malicious purposes. We send notices to the site owner and hosting provider in the former case, and to the hosting provider and domain name registrar in the latter. Figure 4 plots the survival function for full notices where Google has not already sent notices themselves. We only consider that case because the prior sections have already shown these notices to be the most effective. The solid green line plots the survival function for compromised sites, whereas the orange line with long dashes shows the survival of purely malicious websites. We can see that compromised sites are consistently cleaned up more quickly than purely malicious sites, but the difference is not statistically significant at a 95% confidence interval.

It can be useful distinguish between the type of malware hosted on a particular URL, such as malicious executables, injected JavaScript or VBScript code, redirects, and exploits. URLs compromised by a malicious executable host the file. By contrast, URLs compromised by injected JavaScript code contain malicious code mixed in with legitimate code.URLs compromised by redirects return either a 301 or 302 HTTP status code where the target is a malware website; sometimes these redirects only occur with a particular HTTP referrer. URLs hosting exploits attempt to infect users' computers through holes in

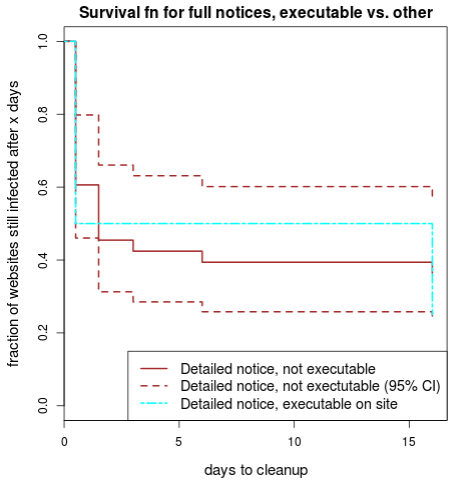


Figure 5: Survival function for cleanup time of websites hosting executables and others.

Adobe Flash, Adobe Reader, Java, Internet Explorer, and so on. For the purposes of our study, we divide URLs by whether they host a malicious executable or not, since this distinction is fairly straightforward.

Figure 5 plots the survival functions for those sites directly hosting executables and for those that do not. There is little discernible difference between the cleanup rates. Thus, we conclude that whether a site delivers executables does not affect the speed at which it is removed.

4 Related work

Researchers have documented the proliferation of web-based malware in recent years. Provos et al. described the pervasiveness of "drive-by-download" targeting web search results [6]. Rajab et al. describe the evolution of attack and defense between web-based malware developers and those attempting to eradicate malware [7].

Moore and Clayton examined a wide range of online content that is subject to "notice and take-down" efforts [5]. By comparing different classes of content, they argued that the incentives for defenders to remove content was the biggest driver for whether or not parties complied with take-down notices. Undoubtedly, incentives can influence whether or not an operator acts on a malware notice such as those sent out in our study. We suspect that website operators who believe that an infection could directly harm themselves and not just their customers (e.g., if they fear punishment in search-engine results) are more likely to take action after receiving a notice. Of course, in our study we do not directly observe incentives of operators; instead, we can only measure the effect of a successful cleanup.

In a study of website take-down companies serving banks targeted by phishing, Moore and Clayton found that a lot more information is being collected on bad websites than is being shared [4]. Our study lends more empirical support to the argument that notices can make a difference, supporting Moore and Clayton's recommendation for security firms to issue more reports.

While we are not aware of any other work describing experiments to assess the effectiveness of malware reports as a security countermeasure, a number of papers have studied the extent to which other interventions could be effective in combating online crime. Levchenko et al. found substantial concentration in the registrars and payment processors used by spam-advertised websites [2]. They recommended that notices of the illicit behavior should be sent to these points of concentration as a way to disrupt the spammers' operations. Liu et al. consider whether it would be effective to notify registrars of known bad domains in the hopes of suspending them [3]. They conclude that criminals are more adept at shifting to new domains than the registrars can act to suspend the offending domains. We have not yet considered how attackers might adapt if detailed notices of malware were to be sent en masse.

5 Conclusions, limitations and future work

We have described an empirical study measuring the impact of sending malware reports to affected stakeholders in the hope of remediating malware. The key takeaway from our study is that notices work. Their impact is immediate – 32% of malware-distributing websites are cleaned within one day of sending a notice, compared to just 13% of sites not receiving a notice. Notices are also long-lasting – 62% of websites receiving a detailed notice were cleaned up after 16 days, compared to 45% of websites not receiving a notice.

There are two important caveats to these findings, however. First, the notices must be detailed – notices that lack detail about the nature of the compromise appear to have no distinguishable impact compared to not sending a report at all. Second, only the first notice is likely to be helpful. Websites that have also received malware warnings from Google were no more likely to respond to our additional reports. Consequently, we recommend that prospective malware reporters first examine whether others have likely already sent notifications before expending valuable resources on constructing detailed reports. Fortunately, reporters can easily check whether a website has been flagged as bad by Google by checking the suspected URL against Google's Safe Browsing API [1]. Alternatively, one could consult a resource such as StopBadware's malware website clearinghouse [8].

There are a number of limitations to the current study

that might be improved in future work. First, we could identify more explanatory variables that could be tested for affecting the malware cleanup rates. Broadly speaking, we could collect relevant data on characteristics of the defender or the malware itself. Defender characteristics might include attributes of the hosting provider (e.g., large vs. small, shared vs. dedicated hosting, country headquarters), site owner (company size, company vs. individual, country headquarters) or associated registrar. Regarding malware characteristics, we already observed a small but not statistically significant difference between purely malicious and infected hosts. We saw no difference between those distributing executables and others. But perhaps we need to draw better distinctions, such as between intermediaries and malware delivery end-points, or between the technology of attack (e.g., JavaScript or browser plugins).

This brings us to another limitation of the current study: the relatively small size of the data set. Reports to BadwareBusters constitute a very small fraction of overall malware observed. While the sample size was sufficient to reach statistically significant results, it would be nice to carry out an experiment on larger samples, particularly if we want to differentiate between malware and defender characteristics as suggested above.

It would also be nice to test empirically whether the reputation of the report sender affects response rates. For example, many website operators are more likely to listen to reports coming from an organization such as Google, where there is a fear of punishment in search results if the report is ignored. Similarly, sending reports from well-known universities or organizations may be handled differently to reports coming from security firms or unknown groups.

A final issue that could be examined in future work is the potential for re-infection. We frequently observed websites that would be cleaned up quickly, only for the infection to re-appear days later. In this study, we treated re-infected sites as not cleaned up. In fact, the operators of the websites may very well be responding to the notice, but they lack the skills or dedication to completely eradicate infection. It would be interesting to investigate what goes wrong on these sites to prevent complete remediation.

Acknowledgements The work was undertaken while the authors were affiliated with Wellesley College, and while the first author was an intern at StopBadware. We are grateful to Maxim Weinstein and Isaac Meister for their input in designing the experimental methodology. We also thank the reviewers and our shepherd, Chris Kanich, for their valuable feedback. Finally, Tyler Moore acknowledges the support of the Norma Wilentz Hess Visiting Assistant Professorship at Wellesley College.

References

[1] GOOGLE. Safe Browsing API, 2011. `https://developers.google.com/safe-browsing/`.

[2] LEVCHENKO, K., CHACHRA, N., ENRIGHT, B., FELEGYHAZI, M., GRIER, C., HALVORSON, T., KANICH, C., KREIBICH, C., LIU, H., MCCOY, D., PITSILLIDIS, A., WEAVER, N., PAXSON, V., VOELKER, G., AND SAVAGE, S. Click trajectories: End-to-end analysis of the spam value chain. In *IEEE Symposium on Security and Privacy* (Oakland, CA, May 2011), pp. 431–446.

[3] LIU, H., LEVCHENKO, K., FELEGYHAZI, M., KREIBICH, C., MAIER, G., VOELKER, G. M., AND SAVAGE, S. On the effects of registrar-level intervention. In *USENIX Workshop on Large-scale Exploits and Emergent Threats (LEET)* (Boston, MA, March 2011).

[4] MOORE, T., AND CLAYTON, R. The consequence of non-cooperation in the fight against phishing. In *Third APWG eCrime Researchers Summit* (Atlanta, GA, October 2008).

[5] MOORE, T., AND CLAYTON, R. The impact of incentives on notice and take-down. In *Managing Information Risk and the Economics of Security* (2008), M. Johnson, Ed., Springer, pp. 199–223.

[6] PROVOS, N., MAVROMMATIS, P., RAJAB, M., AND MONROSE, F. All your iFrames point to us. In *17th USENIX Security Symposium* (Aug. 2008).

[7] RAJAB, M. A., BALLARD, L., JAGPAL, N., MAVROMMATIS, P., NOJIRI, D., PROVOS, N., AND SCHMIDT, L. Trends in circumventing web-malware detection. Tech. Rep. rajab-2011a, Google, July 2011.

[8] STOPBADWARE. Badware website clearinghouse, 2011. `http://www.stopbadware.org/home/clearinghouse`.

[9] STOPBADWARE. Best practices for reporting badware URLs, Oct. 2011. `http://www.stopbadware.org/pdfs/best-practices-reporting-badware-urls.pdf`. Last accessed March 16, 2012.

[10] TURNBULL, B. W. Nonparametric Estimation of a Survivorship Function with Doubly Censored Data. *Journal of the American Statistical Association 69*, 345 (1974), 169–173.