Make notifications great again: learning how to notify in the age of large-scale vulnerability scanning

Cetin, F.O.; Ganán, Carlos; Korczynski, Maciej; van Eeten, Michel

Publication date
2017

Document Version
Accepted author manuscript

Published in
16th Workshop on the Economics of Information Security (WEIS 2017)

Make Notifications Great Again: Learning How to Notify in the Age of Large-Scale Vulnerability Scanning

Orcun Cetin, Carlos Gañán, Maciej Korczyński, Michel van Eeten
Delft University of Technology, the Netherlands
Email: {f.o.cetin, c.hernandezganan, maciej.korczynski, M.J.G.vanEeten}@tudelft.nl

***Abstract*—As large-scale vulnerability detection becomes more feasible, it also increases the urgency to find effective large-scale notification mechanisms to inform the affected parties. Researchers, CERTs, security companies and other organizations with vulnerability data have a variety of options to identify, contact and communicate with the actors responsible for the affected system or service. A lot of things can – and do – go wrong. It might be impossible to identify the appropriate recipient of the notification, the message might not be trusted by the recipient, it might be overlooked or ignored or misunderstood. Such problems multiply as the volume of notifications increases. In this paper, we undertake several large-scale notification campaigns for a vulnerable configuration of authoritative nameservers. We investigate three issues: What is the most effective way to reach the affected parties? What communication path mobilizes the strongest incentive for remediation? And finally, what is the impact of providing recipients a mechanism to actively demonstrate the vulnerability for their own system, rather than sending them the standard static notification message. We find that retrieving contact information at scale is highly problematic, though there are different degrees of failure for different mechanisms. For those parties who are reached, notification significantly increases remediation rates. Reaching out to nameserver operators directly had better results than going via their customers, the domain owners. While the latter, in principle, have a stronger incentive to care and their request for remediation would trigger the commercial incentive of the operator to keep its customers happy, this communication path turned out to have slightly worse remediation rates. Finally, we find no evidence that vulnerability demonstrations did better than static messages. In fact, few recipients engaged with the demonstration website.**

I. Introduction

The Internet's decentralized and trans-boundary architecture requires effective voluntary collaboration between defenders to fight off security threats. This can take the form of abuse reporting, where one party notifies another of an abuse incident and asks it to act against the abuse. Another important collaborative mechanism is to detect and remediate vulnerabilities before they are exploited by notifying the entity responsible for the vulnerable system or service.

Notifications that drive such voluntary cyber-defense take on many forms, from manually crafted emails sent to webmasters all the way to machine-generated feeds that recipients can tailor to their information needs. Some notifications are unsolicited and pushed to recipients, others require the recipients to take action and request data via APIs or mechanisms. Despite differences in the content, context and technology of how the countermeasures are deployed, each is premised on some type of notification about the abuse or vulnerability, being sent from one party to another.

In this paper, we focus on vulnerability notifications. They have been around for quite a while. The security community, however, has only recently started to study the effectiveness of these mechanisms. We know remarkably little about the aspects and factors that drive higher vulnerability remediation rates and how recipients feel about various types of notifications [1], [2]. Moreover, there is a lack of evidence-based guidelines on how to make large-scale notification mechanisms more useful and effective in remediating vulnerabilities.

Any large-scale notification mechanism will have to decide on a variety of issues regarding how to get the vulnerability information in the right hands and how to incentivize actual remediation. In this paper, we investigate three issues: What is the most effective way to reach the affected parties at scale? What communication path mobilizes the strongest incentive for remediation; contacting the nameserver operator directly, their customer or the network operator? And finally, what is the impact of providing recipients a mechanism to actively demonstrate the vulnerability for their own system, rather than sending them the standard static notification message. We study these questions by undertaking several large-scale notification campaigns for authoritative nameservers that are vulnerable to so-called "zone poisoning" [3].

In the next section, we outline the methodology used for this experiment. The results of the experiment are explained in Section III. In Section IV, we present an explanatory analysis of email bounces and remediation. We explore reactions of email recipients in section V. Finally, we compare our findings to the related work in section VI and we summarize our conclusions in sectionVII.

II. Methodology

We designed an experiment around nameservers that are configured to allow non-secure dynamic updates. This allows for an attack called *zone poisoning*. In this section, we explain the overall design of the study, which is summarized in Figure 1. First, we briefly describe the vulnerability and how we identify vulnerable nameservers. Then we outline the three notification campaigns using different communication channels: nameserver operators, domain owners and network

operators. Subsequently, we discuss the experimental design that was used in each campaign to test the impact of the different notifications. We describe content of the notifications, the demonstration website and the recipient survey. Fourth, we describe our rationale for constructing the experimental groups. Fifth, we discuss the ethical issues associated with our approach. Finally, we explain how we evaluate the results.

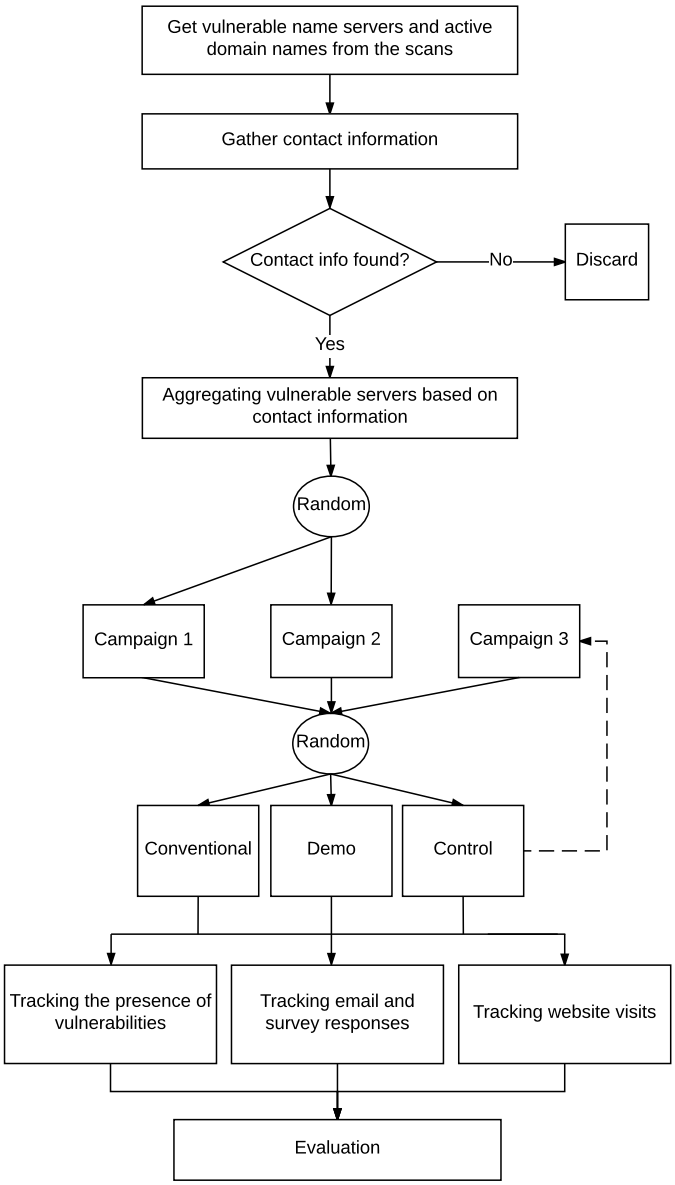


Fig. 1: Flow diagram of the progress through the phases of our experiment

A. Vulnerability: Non-secure DNS Dynamic Updates

Korczyński et al. presented a measurement study of authoritative nameservers that allow non-secure dynamic updates [3]. This vulnerable configuration allows anyone on the Internet to freely manipulate DNS entries in the zone files of that authoritative nameserver. This attack is referred to as *zone poisoning*. The attack is as simple as sending a single DNS dynamic update packet that is compliant with RFC 2136 [4] to a non-secure server. In the simplest version of an attack, a miscreant could replace an existing `A` or `MX` resource record in a zone file of an authoritative server and point the domain name to an IP address under control of an attacker. This can be used, for example, for phishing or for intercepting email by changing the record for `mail.domain.com`. The requirements for the attack to succeed are: non-secure updates are allowed by an authoritative server for a given zone and the miscreant knows the domain name and its nameserver. Finding this information is trivially easy. In short: it is a serious security threat. The original paper [3] discusses the threat model in more detail.

Korczyński et al. analyzed a random sample of 2.9 million domains and the Alexa top 1 million domains and found that at least 1,877 (0.065%) and 587 (0.062%) of domains are vulnerable, respectively. Among the vulnerable domains were governments, universities and banks, demonstrating that the threat impacts important services.

The first measurement study was extended from a sample to a comprehensive scan of the domain name space. Between September 21 and October 11, 2016, Korczyński et al. performed a global scan of the non-secure DNS dynamic updates and found 309,687 vulnerable domains and 5,738 IP addresses with vulnerable authoritative nameservers. In total, they counted 579,186 unique "domain, nameserver" tuples. Here, we limited our study to the 21,506 domains that were active during the period of our experiments – which corresponds to 4,149 IP addresses of the vulnerable nameservers with Start of Authority (SOA) records.

B. Experiment

In this section, we outline the research questions which we attempted to answer via the experiment. We were specifically interested to answer following research questions:

1) How Can You Reach Resource Owners at Scale?

Security researchers have a variety of options to identify the contact details of owner, operator or user of the vulnerable resource. One approach is to use dedicated mail aliases as mentioned in the RFC 2142 for abuse and network-related problems [5]. For DNS-related problems, the RFC says to use the SOA `RNAME` field to provide contact information for the zone's administrator. Moreover, the RFC defines "hostmaster" as the mail alias to be used for DNS issue. It also mentions "abuse" as the email aliases that can be used for generic abuse and vulnerability notifications.

During the first campaign, we test the effectiveness of reaching administrators of vulnerable nameservers by sending a notification to the email as specified in the SOA `RNAME` field. When this field was not present, we used the "abuse" email alias.

During the second campaign, notifications were sent to the owners of vulnerable domains. We obtained the contact details from the registrant's email address in the WHOIS records of the domain. When we couldn't find the registrant's email address, we sent the notification to `<hostmaster@domain>`.

Furthermore, the "abuse" email alias for domain was used as a fallback option when a bounce report of the initial notification was received.

2) Which Channel Contains the Strongest Incentive for Remediation?

Next to getting the notification to the chosen recipient, there is also the issue of whether that recipient has an incentive to perform remediation. Since there are different affected parties that could be notified of this vulnerability, we wanted to test whether it was more effective to contact resource owners directly, to go via their customers or to go via their network operators. The direct route seems the most obvious communication channel, but the name server operator might not have an incentive to remediate. The domains threatened by zone poisoning might not be his. Changing the configuration to a secure mechanism for dynamic updates might also generate cost, for example, to replace this functionality with what is inevitable a more complicated solution than the non-secure configuration. Under these conditions, it might be rational to wait and see whether actual abuse will occur and with what frequency.

The domain owners, which are typically the customers of the nameserver operator, might care more about protecting their domain. Our notification suggested that they might have to contact the nameserver operator, for example their hosting provider, to ask for the problem to be remediated. The operator probably has a stronger incentive to act on such a customer request than on the friendly advice of an academic research team. We tested which path leads to better remediation by contacting different recipients in each of the three campaigns. First, we notified nameserver operators directly via SOA `RNAME` field. Second, we contacted domain owners via the registrant's email address in the domain `WHOIS` record. Third, we would notify the next higher level intermediary, the network operator, via `IP WHOIS` abuse contact field.

3) Does a Demonstration of the Vulnerability Produce Better Remediation?

Since recipients might receive many vulnerability notifications, something that will only increase with the rise of large-scale vulnerability detection, they are probably not willing or able to act on all of them. It seems inevitable that recipients some form of triage the incoming messages, if only in the form of ignoring those that do not seem trustworthy, credible or critical.

Providing recipients with a simple way to demonstrate the vulnerability for their own nameserver or domain, would allow them to immediately verify the trustworthiness, credibility, and criticality of the notification. To test whether this improved remediation, we built a website that demonstrated the vulnerability. Recipients could let the site inject a harmless record in the zone file of a vulnerable domain. The site would show the existence of this new DNS record, proving that anyone on the Internet could change any DNS record for that domain. (We included controls to avoid abuse, recipients could only test their own domains or nameservers.)

To test whether the demo makes a measurable difference, we designed two different treatments: one standard notification message and one notification message that included the same information plus a link to a website that we built that demonstrated the vulnerability. In each of the three campaigns, the recipients were assigned to one of three groups: a control group that received no notification; a treatment group receiving a conventional notification; and a second treatment group receiving a notification message with a link to a site we built where the vulnerability is demonstrated. We discuss the notification content and website in more detail in subsection II-D.

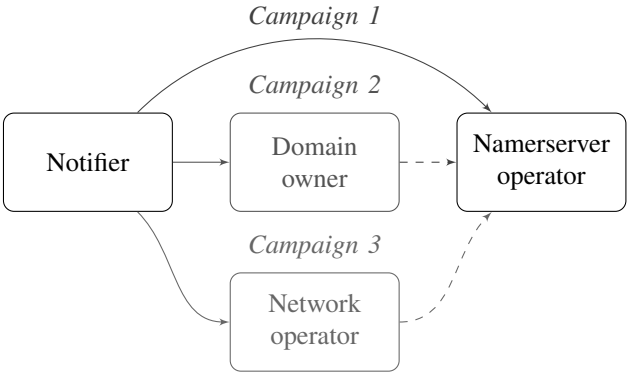


Fig. 2: Communication channels per campaign

C. Group Assignment

There are several steps in the overall experiment, as illustrated in Figure 1. It is a bit complicated, but the easiest way to think about is this: the experiment with the two notification treatments (notification with and without access to the demonstration website) is repeated three times, once for each communication channel (see Figure 2).

We chose for three sequential, rather than parallel, campaigns to keep the experiment manageable and to prevent possible contamination. Sequential campaigns could have cause contamination in various ways. For example, if the first 2 campaigns would have run sequentially, once we contacted a domain owner, then she might have contacted the nameserver operator, as we hope she would. The operator, however, might be responsible for other nameservers or domains as well, which might be in another treatment group or in the control group in the other campaign. As a result of this, same nameserver operator will appear in different treatment groups, thus receiving different treatments.

There are several assignment processes during the study. The process starts with identifying the relevant contact point. For each vulnerable nameserver, we extracted the email address of the person or organization responsible for the DNS zone from the corresponding SOA record. In 256 (out of 4518) cases, the SOA record for the nameservers was not present, hence we removed the nameserver and associated domain names from our study. Next, we aggregated the nameservers and domain names by unique SOA contact information. This resulted in 3967 unique nameserver contacts. We then ran-

domly assigned each contact to the first or second campaign (see Figure 2).

For the first campaign, the nameserver operators were randomly assigned to one of three groups: control, conventional notification and demonstrative link. The contacts assigned to the control group received no notification during this campaign. As we discuss below, we did notify them later on in the study. The measurement period of the first campaign lasted 19 days. We tracked remediation, survey and email responses and website visits.

Once the first campaign was done, we moved to the second campaign. First, we checked whether the domains and nameservers assigned to this campaign were still vulnerable and found that 70 (out of 1984) cases were remediated without being notified by us. And we also checked whether units assigned to second campaign shared any IP address or domain with the previous campaign and found that 451 (out of 1914) cases were sharing at least one IP address or domain. They were removed from the experiment. To identify the relevant contact information, we had purchased `WHOIS` data[1] and extracted the registrant's email address. We did not use any other email address field in the `WHOIS` record, as they could lead to the hosting provider or another entity. When registrant's email was missing, we generated an email address using the "hostmaster" email alias, as recommended by RFC 2142 [5].

Next, we conducted one more aggregation. If two nameservers had different names but they both resolved to the same IP address, then we bundled them, and the associated domains, together. This was done to further reduce the risk of contamination. We then randomly assigned each unique nameserver contact point (or bundle thereof) to one of three groups: control, conventional notification and notification with link to demonstration website. All domains associated with a nameserver contact would receive the corresponding treatment assigned to that contact. For example, for all domains that ended up in the conventional notification treatment group, we contacted the registrants with the conventional notification message. It is important to reiterate that in the second campaign we did *not* contact any nameserver operator directly.

Once the measurement period of the second campaign ended, we took the control groups of the first and second campaign as the subjects for the third campaign. First, we checked that the domains and nameservers were still vulnerable. As it turns out that 95% of these hosts were still vulnerable. We extracted contact information for the network operators by querying Abusix's Abuse Contact DB [6] for the `IP WHOIS` abuse contact that belongs to the IP address of the vulnerable nameserver. These abuse contacts belong to upstream intermediaries, such as ISPs and hosting companies. Next, we aggregated the vulnerable domains and nameservers per unique abuse contact point. We then randomly assigned these contacts to one of the three treatment groups, as was done in the first and second campaign.

[1]We purchased `WHOIS` domain data from whoisxmlapi.com.

D. Notifications, Demonstration Website and Survey

Notifications for both treatments were sent from the same dedicated email account belonging to Delft University of Technology. To reduce the risks of unsuccessful email transmission, we disabled inbound and outbound spam filters used by the university.

The conventional notification treatment consisted of an email with a plain text vulnerability report. It contained a brief explanation of how we discovered the vulnerability, what the security impact is if it is abused, and how it can be remediated. We enumerated the vulnerable nameservers or domains associated with the contact point. The message concluded with a link to a short survey. The other treatment consisted of basically the same notification, plus a link to the vulnerability demonstration website. Full details of the notification messages can be found in Appendix A.

We built and operated the demonstration website. Figure 7 shows screenshots of the interface. The site provided recipients with an opt-in tool that would provide a live demonstration of the vulnerability for their nameserver or domain – that is, an actual record, albeit a harmless one, would be injected into the zone file. The new record added a subdomain called *zonepoisoning* to the vulnerable domain. This sub-domain would then correctly resolve through DNS and point to a webserver belonging to our experiment, showing that the record was successfully inserted. The added record remained in the zone file for 10 minutes, after which it was removed automatically. After every interaction, website interface shows the results of the subdomain injection attempt. Vulnerable servers trigger an interface where a link to created subdomain and an explanation is displayed to verify the existence of the vulnerability (see Figure 8a). On the other hand, patched servers triggered a different interface, explaining the unsuccessful injection attempt (see Figure 8b).

The website, the server to which the new subdomain resolved and the server used for the scans for vulnerable nameservers and domains all provided information on how to opt out of our study. To prevent potential abuse, we provided recipients with a link containing a unique token that allowed us to restrict what domains or nameservers could be tested by the visitor of the website. Recipients could only demonstrate the vulnerability for the nameserver, domain or networks for which they were the contact point.

The website and the notifications included a link to a short survey where the recipients were asked to answer several questions about our notification process. The questionnaire was designed to capture the recipients' reaction to our notifications, to notifications in general and to the way we conducted our research.

E. Tracking Process

To track remediation during each campaign, and to update our data on vulnerable nameservers and domains, we performed 7 scans between November 3 and December 29, 2016.

We used the scanner that was developed by Korczyński et al. [3]. It sends a DNS update request packet that is compliant

with RFC 2136 [4]. The request was to add an extra `A` record to the zone file, associating a new subdomain (e.g., `researchdelft.example.com`) with the IP address of the web server of our project. When a nameserver operator would visit the IP address, she would encounter a page with an explanation of the study and an easy opt-out mechanism (see Figure 9).

Our scanning setup was designed to have minimal impact, while also taking into account random packet losses. We first sent two DNS update request packets. We then performed four DNS lookups, from two different measurement servers, to verify if the added domain correctly resolved to our web server's IP address. Next, we removed the test DNS record by sending a delete update request. Finally, we queried the authoritative DNS server and try to resolve the subdomain once more, in order to confirm that the added record was successfully deleted.

We considered an authoritative nameserver as remediated if it no longer appeared vulnerable in any subsequent scan. A domain was considered as remediated if none of its authoritative nameservers are found to be vulnerable.

F. Ethical Considerations

Our study aims at improving the deliverability of vulnerability information to owners of computing resources, such as websites or servers. Vulnerability notifications are a well-established practice to help operators of vulnerable resources to better protect themselves against criminals who might abuse the vulnerabilities.

The only valid method available to detect and demonstrate the vulnerability was to insert a benign record into a zone file. We weighed the tradeoffs and decided that the benefit of helping the server operators to protect themselves outweighed the potentially intrusive nature of the scans. The ethical considerations are discussed in more detail in [3]. The inserted records were only present for a very short time. We did not interact with any of the existing records in the zone. We did not observe or hear about any problems with the vulnerable servers because of our scans, as we expected, since our interaction with the servers was fully compliant with the relevant standard. Furthermore, recipients were provided with an opt-out mechanism in every engagement. During the study period, only one recipient asked to be excluded of the study.

G. Evaluation

To assess which communication channel contains the strongest incentives for remediation, we evaluate the results based on two metrics: (i) reachability, i.e., the email bounce rate; and (ii) the remediation rate. We measured the impact of the vulnerability demonstration by comparing the remediation rate of the recipients who visited and/or used the demonstration tool versus those who did not. In addition, we explored the email and survey responses to learn more about how various recipients perceived our vulnerability demonstration website and the content of the notifications.

III. Notification Results

In the previous section, we outlined the experimental design, methodology and objectives. In this section, we present the results of each campaign on the deliverability of notifications and on the remediation rate. Next, we discuss the efficacy of the demonstration website. We end with a comparative analysis of the communication channels.

A. Notification Deliverability

In this section, we analyze the deliverability rates of the notifications. Table I summarizes the bounce rates per campaign.

1) First Campaign: Reaching the relevant contact points at scale turned out to be a huge problem. As shown in table I, initially 669 emails were sent with a link to demonstration website and 657 emails with conventional content. Of these 669 emails for the demonstration group, 70% returned a delivery failure. Similarly, 67.73% of the emails with conventional notifications failed to be delivered.

To reach more affected parties, we sent a second email when the first one had generated a failure. This second email was sent to an address we generated in compliance with RFC 2142 [5], of the form `<abuse@domain.com>`, where the domain corresponded to the nameserver domain. So the operator of `ns1.example.com` would be contacted at `<abuse@example.com>`. We sent an additional 692 emails this way: 335 for the conventional treatment group and 357 for the demonstration group. This second attempt incurred an even higher bounce rate: on average, 84% of these messages generated a delivery failure.

2) Second Campaign: We sent 2,051 emails to domain owners in the second campaign: 1,111 emails to the conventional notification group and 940 emails to the demonstration group. Of these 2,051 emails, 39.78% bounced on average. The rate was slightly higher for the demonstration group. Similar to the first campaign, when a notification could not be delivered, we applied a fallback option. We sent a second email to `<abuse@domain.com>` addresses for vulnerable domains. In total 561 emails were sent in hope to reach more vulnerable domain owners. Around 89% of these bounced also.

3) Third Campaign: We sent 417 emails during the third campaign. For network operators, as identified via the `IP WHOIS` abuse contact for the IP address of the vulnerable nameserver, reachability was much better. Only 36 out of the 417 notifications generated a delivery failure. For this reason, we did not use a fallback option.

B. Remediation Rates

The reachability of nameserver operators was poor, for domain owners it was slightly better and for network operators it was quite good. This raises the question of whether this is also connected to a difference in remediation. In this section, we analyze the remediation rates of the different treatment and control groups for each campaign.

Table II provides a summary of the status of the vulnerable servers during three different measurements in the first and second campaign, and during two measurements for the third

Campaign	Treatment type	Total number of aggregated contacts	Number of emails initially send	Rate of undelivered emails	Number of fallback emails send	Rate of undelivered emails
1	Demonstration	669	669	70.40%	357	82.07%
	Conventional	657	657	67.73%	335	86.26%
2	Demonstration	451	940	44.68%	279	88.88%
	Conventional	451	1111	35.64%	282	89.00%
3	Demonstration	184	208	12.01%	–	–
	Conventional	183	209	5.2%	–	–

TABLE I: Bounce rates

Treatment Type	Campaign 1				Campaign 2				Campaign 3		
	#	After 3 days	After 13 days	After 19 days	#	After 3 days	After 13 days	After 19 days	#	After 3 days	After 13 days
Control	657	3.04%	4.26%	5.02%	476	2.31%	3.78%	4.62%	320	0.3%	1.87%
Demonstration	267	12.35%	14.23%	18.35%	345	5.50%	6.37%	10.14%	382	4.97%	8.11%
Conventional	260	8.84%	9.61%	14.23%	327	5.81%	6.72%	12.23%	329	3.03%	5.77%

TABLE II: Summary statistics remediation per treatment group, counted per unique SOA contact points

campaign. The additional third measurement for the first two campaigns allows us to see the impact of the fall-back notifications. The table reports the percentage of contact points that took action, excluding those that we could not reach. Overall, remediation rates were low. The highest rate for any group or campaign was 18% of all vulnerable nameservers.

1) First Campaign: Notification clearly makes a difference. In the control group, 5% of the contact points remediated the vulnerability within 19 days, compared to 18% and 14% for the two treatment groups. We did a log-rank test and found that difference between the treatment groups and the control group is significant ($\chi^2 = 41.1$, $p = 1.44e{-}10$), while the difference between the two treatments is not ($\chi^2 = 1.8$, $p = 0.182$). In short, the demonstration did not make a difference.

2) Second Campaign: The pattern for the second campaign is similar: notifications increase remediation, compared to the control group (log-rank test: $\chi^2 = 41.1$,$p = 1.44e - 10$), but there is no significant difference among two treatments ($\chi^2 = 1.8$, $p = 0.182$). The remediation rates turned out to be slightly lower when contacting nameserver operators via their customers, compared to the first campaign, where we contacted them directly: 11% versus 16%, on average.

3) Third Campaign: Since the third campaign focused on abuse contacts at network operators, we aggregated the vulnerable nameservers per network operator. Table III mentions the remediation rate per recipient. Note that these numbers are different from Table II, as the latter standardized all rates on unique SOA contact points, to make the number comparable. The pattern is basically the same as for the first two campaigns. After 13 days, 15% of the demonstration and and 8% conventional notification groups achieved respectively. Again, a log-rank test concluded that control and treatments were significantly different, while the treatment groups were not.

There are two key findings from these remediation rates. First, providing a vulnerability demonstration to recipients had no observable impact on remediation for any of the contacted parties. Second, there is a modest, but significant difference between the direct and indirect communication channels by comparing the percentage of contact points that took action (see table II). Figure 3 plots the survival probabilities. The remediation rate of the first campaign, which contacted the nameserver operator directly, was slightly higher than during the two indirect campaigns (log-rank test:$\chi^2 = 5.2$, $p = 0.022$).

Treatment Type	#	After 3 days	After 13 days
Control	183	0.54%	3.27%
Demonstration	164	9.75%	14.63%
Conventional	173	5.78%	8.09%

TABLE III: Percentage of remediation by network operators in third campaign

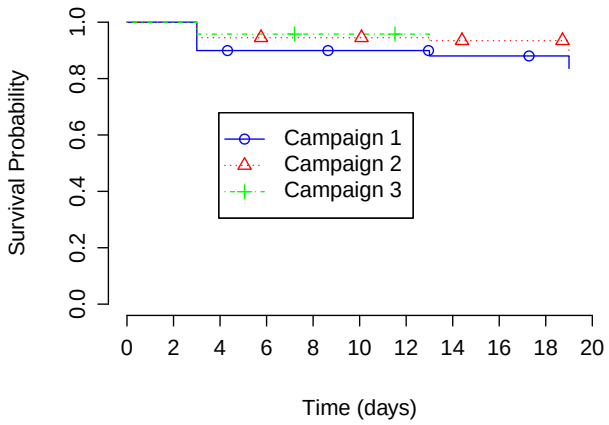


Fig. 3: Survival probabilities across the campaigns

C. Efficacy of the Demonstration Website

As a part of our experiment, we built a website that could be used by recipients to demonstrate the vulnerability for their own nameserver or domain. We had two slightly different versions for domain owners and nameserver operators, so

that we could tweak the language to their situation. The version for nameserver operators also contained more technical information to assist with the remediation process (see Figure 7).

During the experiment period, we tracked the visitors of both websites. As it turns out, most recipients of the link did not visit the website. The number of visitors is presented in Table IV. In the first campaign, only 12.2% of the operators visited the website. Those that did made 192 injection attempts. Only about half of these attempts were successful in adding a record. The rest of the attempts failed because the visitor tried to inject a domain name that was not associated with their vulnerable nameserver. In the second campaign, only 7.07% of the domain owners who received the link visited the website. Visitors used the demonstration website 81 times in total. Unlike the previous campaign, 82.71% of the injection attempts were successful. The third campaign showed a similar picture: only 14.75% of the recipients visited the site, they made 137 attempts of which 64.23% were successful.

We have no good explanation for why visitors failed so often to demonstrate the vulnerability. To some extent, this is probably trial and error driven by curiosity. Some of the failed attempts, however, reveal usability problems. While we thought we had designed a very simple interface with straightforward instructions, user behavior told us otherwise. The nameserver operators often tried to test nameserver names, rather than the domains of which the zone file was vulnerable. This happened even though the site instructed otherwise, and we supplied them with a full list of domains to test in the notification email and even proposed a specific domain to test in the main part of the text. All in all, this is a painful lesson that it is very easy to underestimate how hard the problem is of usability of user engagement in the area of security. We can add this lesson to the growing body of work in this area [7].

	Campaign 1	Campaign 2	Campaign 3
Number of visitors	32	39	27
Number of attempts	192	81	137
Number of successful attempts	104	67	88
Number of failed attempts	88	14	49

TABLE IV: Summary statistics on demo website visits

To analyze whether the website helped visitors to expedite remediation, we compared remediation rates of visitors and non-visitors. Figure 4 plots the survival probabilities for both groups. Figure 4a shows that after 3 days more than 40% of visitors had taken action, while those who did not visit had remediated less than 10%. After 19 days, almost 60% of the visitors took action, while the non-visitors still hovered around 10%. The same pattern emerged during subsequent campaigns. Log-rank tests show that these differences are significant. We have no hard evidence on what caused the higher remediation rate. The site may have helped, but it is more likely the effect of self-selection. The recipients that were interested in the demonstration website were probably already more willing to act upon the notification.

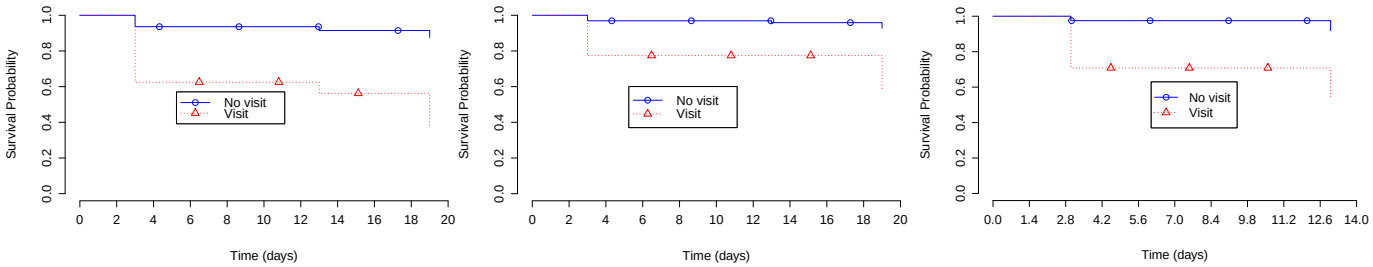
IV. Explanatory analysis

We wanted to get a bit more insight into two of the findings of our experiment: the many delivery failures in contacting affected parties and the low remediation rate. For each issue, we discuss several factors and then feed them into a multivariate logistic regression model to analyze their impact.

A. Modeling Notification Bounce Occurrence

Over the study as a whole, we sent out 5,051 email notifications. Of these, 2,819 triggered delivery failures, a 55.81% bounce rate. We wanted to see if we could explain the probability of a bounce from the features of the recipient's email addresses. We created variables to capture these features.

- *Email Source*: This categorical variable captures the method by which the recipient's email address was obtained. It takes four different values:
  - x_1: **SOA** : This value represents those notification recipients whose email addresses was obtained by digging the SOA record of the vulnerable nameserver and then extracting the `RNAME` field which contains the email addresses of resource owners.
  - x_2: **Domain WHOIS**: This value was set to `TRUE` when the email address was obtained by querying the appropriate `WHOIS` databases corresponding to the gTLD and ccTLD of the vulnerable domains. We then obtained the domain `WHOIS` registrant email field to reach domain owners.
  - x_3: **IP WHOIS**: This value corresponds to those notification recipients whose email addresses were obtained by querying the Regional and National Internet Registry's `WHOIS` databases. We gathered contact details of the entities managing the IP addresses of the vulnerable nameservers.
  - x_4: **Self-generated**: When no contact information was obtained using the aforementioned sources or given information is inacccurete, we generated a RFC-compliant email (i.e., `<abuse@domain>` or `<hostmaster@domain>`).
- x_5: **Privacy-protected Email**: This binary variable is set to `TRUE` when the email address in the `WHOIS` record is behind a proxy service. WHOIS privacy and proxy services are organizations that wish to keep certain information from being made public via WHOIS records [8]. These services can be offered by registrars or their affiliates and they are subject to obligations such as publishing a contact point to receive and distribute notifications. Usually, these services create a random and unique email address for their customers, using their brand suffix. This is entered into the Private Registration Address field of the WHOIS record. Thereafter, when messages are sent to that email address, these services forward the messages to the email address customer listed in their internal registration data. In our dataset, these services are observed for both domain and IP WHOIS



(a) Campaign 1 (b) Campaign 2 (c) Campaign 3

Fig. 4: Survival probabilities for demonstration website visitors vs non-visitors

records. We consider an email addresses to be privacy protected, if the suffix of the address corresponds to one of 17 privacy-protection services we identified.

- x_6: **Free Email**: We consider an email address from a free email provider when the domain name of the email address matched with a list of free email providers (publicly available in [9]). This list contains both currently active and defunct providers. We hypothesize that having a free email account reduces the probability of a bounce, because the same email address could also be being used as a personal email.

We used these variables to model the probability that a notification bounces. A multivariate logistic regression analyses was carried out to assess the influence of each variable. Logistic regression does not restrict the type of variables that can be used. They can be continuous, discrete or a combination of the two. Additionally, the variables do not necessarily have to have a normal distribution. The binary logistic regression equation is:

$$logit(\pi_b) = log\left[\frac{\pi_b}{1-\pi_b}\right], \quad (1)$$

where π_b is the occurrence probability of an email to bounce within the range [0, 1] and can be estimated as:

$$\pi_b = \frac{exp(\beta_0 + \sum_i \beta_i x_i)}{1 + exp(\beta_0 + \sum_i \beta_i x_i)}, \quad (2)$$

where x_i $(i = 1, \ldots, 6)$ refers to the explanatory variables; β_i is the partial regression coefficient; and β_0 is the intercept. $exp(\beta_i)$ is an odds ratio, which mirrors the strength of the correlation between the explanatory variables and the bounce probability. When $exp(\beta) > 1$, a positive correlation exists between the variables and the occurrence probability. When $exp(\beta) < 1$, a negative correlation exists. When $exp(\beta) = 1$, the variables are not correlated with the event.

The results are presented in Table V. All variables have a significant effect on the bounce rates.

Coefficients in logistic regression models can be interpreted as odds-ratios. By calculating the odd ratio from the estimated coefficients, we observe that:

- Contacting affected parties using self-generated email addresses based on RFC standards increases the odds of delivery failure by 588% (odds ratio : 6.88, confidence interval: [6.05, 7.86]).

	Dependent variable:
	bounced
x_1 : SOA	0.794***
	(0.061)
x_2 : whoisDom	−1.752***
	(0.091)
x_3 : whoisIP	−2.333***
	(0.175)
x_4 : selfGenRFC	1.929***
	(0.067)
x_5 : whoisprotection	0.698**
	(0.272)
x_6 : freemail	−1.109***
	(0.227)
Observations	5,051
Log Likelihood	−2,175.878
Note:	*p<0.1; **p<0.05; ***p<0.01
	Standard errors in brackets

TABLE V: Coefficients of the logistic regression model for email bounce occurrence

- Contacting resource owners by using addresses from the SOA record `RNAME` field increases the odds of delivery failure by 121% (odds ratio: 2.21, confidence interval: [1.96 , 2.49]).
- Using the abuse email field of `IP WHOIS` records for notifications decreases the odds of bouncing by 90% (odds ratio: 0.09, confidence interval: [0.06 , 0.13]).
- Using a privacy or proxy services doubles the probability of the email to bounce (odds ratio: 2, confidence interval: [1.15 , 3.36]).
- Contacting addresses from free email providers, as found in `WHOIS` records and SOA `RNAME`, decreases the bounce occurrence by 67% (odds ratio: 0.32, confidence interval: [0.20 , 0.50]).
- Using an addresses gathered from domain `WHOIS` records decrease the bounce probability by 82% (odds ratio: 0.17, confidence interval: [0.14 , 0.20]).

As we hypothesized, contacting affected parties via addresses from `WHOIS` records reduces the odds of a bounce. If

this address is from a free email providers, this further reduces the bounce probability. On the other hand, recipients that are behind privacy-protection services have a significantly higher bounce rate, even though these emails are also gathered from `WHOIS` records.

By far the worst performing in terms of deliverability are self-generated email addresses compliant with RFC recommendations. This mainly indicates that very few nameserver operators and network providers actually follow the recommendations. Many domain owners and DNS services providers (or owners) do not correctly format SOA records, nor integrate mailboxes for security and operational needs.

We assess the goodness-of-fit of our model by calculating the Receiver Operating Characteristic Curve (ROC). The ROC summarizes the model performance between true positive (TP) and false positive (FP) error rates. Figure 5 shows the ROC curve of the model. The area under the ROC curve (AUC score) adds to a combined sensitivity and specificity of 85%. This indicates a good discrimination power of our model when predicting an email will bounce based on the six explanatory variables.

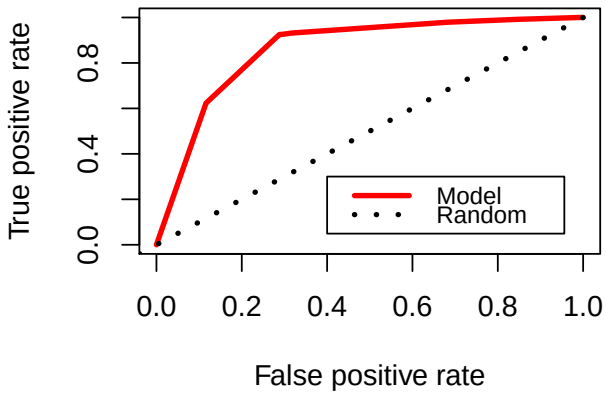


Fig. 5: Logistic regression diagnostic with ROC curve

B. Modeling Remediation Occurrence

We now turn to remediation. We model the chance of remediation as a function of certain features of the nameserver. We derived five variables that might affect remediation:

- *Communication Channel*: This categorical variables represents the type of channel used to reach the nameserver operator. In our experiment we had three different communication channels:
  - x_1: **Direct Channel**: This channel was used during the first campaign as the recipient of the notification was the nameserver operator.
  - x_2: **Indirect Channel Through Domain Owner**: This channel was used during the second campaign where the customers of the nameserver operator were the recipients of the notifications.
  - x_3: **Indirect Channel Though Network Operator**: This channel was used during the third campaign where the notification were sent to the handler of the nameserver IP address.
- x_4: **Number of Vulnerable Domains**: Count of vulnerable domains under a specific nameserver as seen in passive DNS data available in DNSDB.
- x_5: **Number of Domains**: Total number of domains under a given nameserver as seen in passive DNS data available in DNSDB.
- x_6: **Domain Popularity**: Logical variable set to `TRUE` when one or more domains under a specific nameserver are in Alexa's one million top-ranked domains.
- x_7: **Link to Demonstration Website**: Logical variable set to `TRUE` when the recipient of the notification received the notification with the link to the demonstration tool.

We used a logistic regression model to estimate the probability of remediation from the aforementioned explanatory variables. Table VI shows the results. We performed a stepwise inclusion of variables per model. As we move from the initial model to the fifth, we aim to improve the accuracy of model's remediation probability prediction. The discrimination power of the model increased as we added new explanatory variables. The fifth model is the final model we use to explain the remediation occurrence.

In the model, the only non-significant factors are the nameserver size and whether the recipient received the link to the demonstration website. We interpret the coefficients as odds ratios. This provides us with the following observations:

- Having nameservers that include popular domains increase the odds of remediation by 83% (odds ratio: 1.83, confidence interval: [1.23 , 2.65]).
- An increase in the number of vulnerable domains on a nameserver has no effect on it being remediated (odds ratio: 1.00, confidence interval: [1.00 , 1.00]).
- Direct notifications increase the odds of nameserver remediation by 332% (odds ratio: 4.32, confidence interval: [2.47 , 8.10]).
- Notifications to domain owners increase the odds of nameserver remediation by 136% (odds ratio: 2.36, confidence interval: [1.33 , 4.45]).
- Notifications to network operators increase the odds of nameserver remediation by 119% (odds ratio: 2.19, confidence interval: [1.23 , 4.15]).

As we see from the results, the size of the nameserver (number of domains) and sending the notification with a link to the demonstration site did not significantly influence the remediation occurrence in the final model. They were already only weakly correlated in the prior models, which explains the sign flips and changes in significance. Although the number of vulnerable domains on a nameserver was statistically significant, it has very small effect on the odds of remediation.

These results also indicate that direct notifications made the highest impact across all variables. It increased the odds

	Dependent variable:				
	Nameserver Remediation Occurrence				
	(1)	(2)	(3)	(4)	(5)
x_1 Direct Channel	1.591***	1.456***	1.470***	1.463***	1.465***
	(0.291)	(0.300)	(0.300)	(0.300)	(0.300)
x_2 Indirect Channel$_1$	1.012***	0.867***	0.885***	0.866***	0.859***
	(0.294)	(0.304)	(0.305)	(0.305)	(0.305)
x_3 Indirect Channel$_2$	0.959***	0.817***	0.803***	0.803***	0.786**
	(0.295)	(0.305)	(0.306)	(0.306)	(0.307)
x_4 Number of Vulnerable Domains					0.004***
					(0.001)
x_5 Total Number of Domains			0.0001**	0.0001**	−0.0001
			(0.0001)	(0.0001)	(0.0002)
x_6 Hosting Popular Domains				0.632***	0.607***
				(0.194)	(0.195)
x_7 Link to Demonstration Website		0.252*	0.236*	0.225*	0.216
		(0.132)	(0.133)	(0.133)	(0.133)
Constant	−3.676***	−3.676***	−3.689***	−3.738***	−3.731***
	(0.271)	(0.271)	(0.271)	(0.272)	(0.272)
Observations	3,956	3,956	3,956	3,956	3,956
Log Likelihood	−965.880	−964.047	−958.999	−954.294	−950.437
Note:					*p<0.1; **p<0.05; ***p<0.01
					Standard errors in brackets

TABLE VI: Coefficients of the logistic regression model for nameserver remediation occurrence

of remediation by 332% compared to 136%–119% for those notifications sent through an indirect channel.

Similarly to the previous model, we assess the goodness-of-fit by calculating the ROC curve and the computing the AUC value. Though the ROC curve for the model (see Figure 6) shows that model can predict slightly better than the random model (with 69% AUC score), it has poor predicting capabilities.

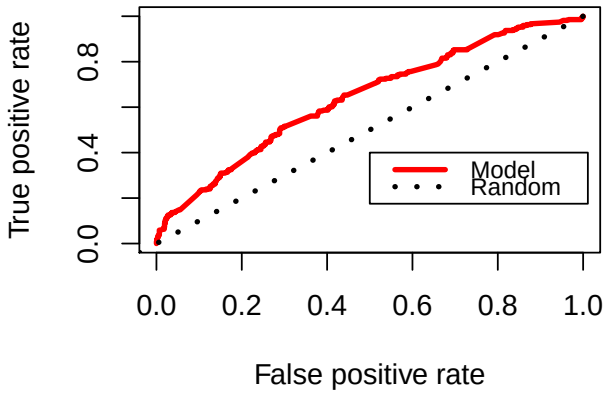


Fig. 6: Logistic regression diagnostic with ROC curve

V. Reactions of Recipients

We observed the reactions of recipients through their email replies and the results of an anonymous survey. All of our notifications included a link to the survey. We received 25 survey responses. This renders the survey useless in terms of understanding the population of recipients. We do discuss the results as anecdotal data that helps to think about how such scans and notification campaigns might be perceived by the affected parties. We received 23 responses to our notifications via our contact email. In following section we analyze reactions of recipients.

A. Survey Responses

The survey was anonymous and each question was optional (see Appendix B for more details). The questions were slightly tailored to the different campaigns and treatment groups. Each survey consisted of 10 questions, with an extra question for the recipients contacted via the indirect channels, asking them whether they could fix the problem by themselves or not.

The survey began by asking demographic-type of questions on the type of organization where the recipient worked and the size of the organization. Next, we asked them whether they had taken any action before getting our notification and whether they were planning to take any action after the notification. These questions are followed by another two questions to learn whether the recipients found it acceptable to scan their nameserver and notify them about the vulnerability. The survey for recipients that received the link to the demonstration website asked about the effectiveness of the demonstration website. Recipients of the conventional notification were asked whether it would have been useful to be provided with a site to safely test and demonstrate the vulnerability. Recipients were also asked whether we notified the right contact point and whether they would like to receive future notifications. At the end of the survey, respondents were given an open question and asked to tell us they wanted about the scans,

notifications or any other issue related to this research or to security notifications in general.

We summarize the results in Table VII. In the first campaign, we received 11 responses, 5 of them from the demonstration group and 6 of them from the conventional content group. Most responses were from hosting providers. The rest of the responders were representatives of DNS service provider, software and gaming company, content delivery network and government. In this campaign, the majority of the responders were from small and medium size organizations and only two responses were from large organizations. In second campaign, where we contact the domain owners, we received 9 responses. Of these, 5 of them belonged to the demonstration group and 4 to conventional notification group. Similarly to campaign 1, small and medium size of organizations were represented more than large organizations. In the third campaign, we received 5 survey responses, 3 of them from the demonstrative notification group and 2 of them from the conventional notification group. The majority of the responders in 3 campaign had large number of employees.

Surprisingly, 9 responders in first and second campaign indicated that they had previously attempted to remediate the problem. After the notification, nearly all responders were planning to remediate the problem. The majority of the responders found our scans and notifications acceptable and they were open to future notifications. Moreover, 23 responders indicated that we reached the correct contact.

In the first campaign, when we asked about the usefulness of the demonstration website, 3 respondents found it very useful and 2 of them found somewhat useful. No one found it not useful. Similarly, the conventional notification group was asked whether it would had been useful if we had provided a demonstration website, 4 of them replied that it would have been very useful and 2 of them replied somewhat useful. In the second campaign, the demonstration website was found very useful for 3 respondents, one did not go the site and one find it not useful. Moreover, half of the responders who belonged to the conventional notification group indicated that providing link to a demonstration website would have been very useful and the other half indicated that providing a link to a demonstration website would have been somewhat useful. In the third campaign, one of the respondents find the demonstration website very useful, other find it somewhat useful and the rest did not go to the site. The respondents for the conventional notifications indicated that providing a link to a demonstration website would have been very useful.

Since in the second and third campaign an indirect channel was used, we wanted to know whether any of the responders in these groups were maintaining the server by themselves. According to these responses, 8 (out of 14) respondents were capable of maintaining the vulnerable server.

B. Email Responses

Throughout the study, we had 23 human replies to our emails (see Table VIII). Two emails stated that the servers in question did not belong to the recipient. Five emails were negative. Two people complained about the scans, one threatened to sue, one claimed to have reported us, not mentioning to whom, and one shared a rather unimaginative insult.

VI. Related work

The effectiveness and feasibility of security notifications has recently become a major concern [10]. Several researchers have begun investigating how security notifications can expedite vulnerability remediation. Li et al. studied the aspects of vulnerability notifications that could increase the vulnerability remediation rates [2]. Their study focused on who to notify and how much information does needs to be included in the notifications. They found that security notifications addressed directly to the owners of the vulnerable resources promote faster remediation than those sent to national CERTs and US-CERT. In addition, their study revealed that detailed vulnerability notifications increased the vulnerability remediation rate compared to terse vulnerability notifications. Stock et al. investigated the feasibility and efficacy of large-scale notification campaigns [11]. Their findings indicated that vulnerability notifications increased the patching rate compared to those that are not notified. However, overall patching rate was marginal. Prior to these work, Kührer et al. conducted a collaborative notification campaign with the CERT/CC and Cisco to notify the network providers and owners of equipment running vulnerable NTP servers [12]. They observed a 92% reduction in vulnerable servers, from 1.6 million to 126,000 in under three months. Regarding the high-profile disclosure of the OpenSSL Heartbleed bug, Durumeric et al. notified operators of detected vulnerable hosts and found that the rate of patching for notified group was 47% higher than the control group [13].

More recently, a number of papers have also started to investigate impact of abuse notifications. Li et al. described in detail the impact of security notifications on 761,935 infected websites that were detected by Google Safe Browsing and Search Quality [14]. They discovered that direct notifications to webmasters via Google Webmaster Console increased the likelihood of cleanup by over 50% and reduced the infection lifetime by 62%. Furthermore, Cetin et al. investigated the impact of sender reputation in abuse reports [1]. Authors found no statistically significant difference between the abuse notifications of senders with varying level of reputation, suggesting that the sender email address does not matter greatly when responding to abuse reports. However, they observed that the notifications resulted in better cleanup than not notifying.

In another previous study, Vasek and Moore conducted an experimental study on malicious URLs submitted to the StopBadware community feeds [15]. They found that abuse notifications sent with detailed information on the compromise are cleaned up better than those not receiving a notice. Surprisingly, they found no difference between the cleanup rates for websites receiving a minimal notice and those not receiving any notice at all.

In two other studies, researchers experimented with web-based malware in hosting services. In a first study, Nappa

Survey Responses	Campaign 1		Campaign 2		Campaign 3	
	Demonstration	Conventional	Demonstration	Conventional	Demonstration	Conventional
Number of participants	5	6	5	4	3	2
Organization type	Hp:5 S/G firm: 1	HP: 2 DNS:1 CDN:1 Government:1	DO: 2 Organization:3	DO: 1 Organization:3	ISP: 2 Private org:1	ISP:1 HP:1
Size (if organization)	1: 1 25-99: 2 100-499: 1 500-999:1	1: 3 2-24: 1 25-99:1 500-999:1	1-24:4 25-99:1	25-99:1 1000+: 2	100-499:1 1,000+:2	2-24:1 100-499:1
Taken Prior Actions	3/2	2/4	1/4	3/1	0/3	1/1
Now Taking Action	4/1	6/0	5/0	4/0	2/1	1/1
Acceptable to Scan	5/0	6/0	3/2	4/0	3/0	1/1
Acceptable to Notify	5/0	5/1	5/0	4/0	2/1	2/0
Demonstration website useful (if provided one)	Very useful:3 Somewhat useful: 2	Very useful:4 Somewhat useful: 2	Very useful:3 Didn't go: 1 Not useful: 1	Somewhat useful: 2 Very useful:2	Very useful:1 Somewhat useful: 1 Didn't go:1	Very useful: 2
Future Notifications	5/0	6/0	4/1	4/0	2/1	2/0
Correct Contact	4/1	6/0	5/0	4/0	2/1	2/0
Can maintain the nameserver	–	–	3/2	3/1	1/2	1/1

TABLE VII: Survey responses

Human Responses	Campaign 1		Campaign 2		Campaign 3	
	Demonstration	Conventional	Demonstration	Conventional	Demonstration	Conventional
Positive Remark	0	1	4	1	5	1
Negative Remark	0	1	3	0	0	1
Neutral Remark	1	0	0	0	1	0
False Positive Notification	2		0		1	

TABLE VIII: Email Responses

et al. sent abuse reports to providers hosting 19 long-lived exploit servers [16]. Only 7 out of 19 providers took action towards cleaning up the malicious servers. In a second study, Canali et al. set up vulnerable webservers on 22 hosting services [17]. They then compromised the webservers and sent out notifications to all hosting providers after 25 days had passed. Approximately 50% took action, generally suspending access to the websites. 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 demonstrates the pitfalls in investigations on abuse reports.

Moreover, Gañán et al. studied how different forms of notifications affected lifetime of ZeuS command and control servers provided by ZeuS Tracker, Cybercrime Tracker and a private company [18]. While ZeuS Tracker and Cybercrime Tracker present a publicly accessible dynamic webpage that displays ZeuS malware command and control servers, the private company did not publicize any of the detected command and control servers. Research concluded that publicized command and control servers were mitigated 2.8 times faster than the ones that were not publicized.

Furthermore, in another study Vasek et al. studied impact of the incident data sharing among Internet operators [19]. Their study concluded that sharing abuse data has a swift effect of cleaning the reported malicious URLs.

Finally, with respect to spam, a quasi-experiment by Tang et al. used two blocklists to compile a large source of e-mail spam and publish aggregated measures on SpamRankings.net[20]. 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 notifying individual hosts in order to remediate infections, the researchers' strategy relied on public shaming. The study indicates that reputation effects could provide an incentive for intermediaries to cooperate in remediating abuse on their networks.

VII. Conclusions

We succinctly state the main results and discuss what they tell us about improving the effectiveness of vulnerability notifications.

A. Reaching affected parties at scale

In light of the rise of large-scale vulnerability scanning, our most sobering result is that there is no good mechanism of getting this wealth of information to the relevant entities. Most of our notifications bounced. Contact information is extremely unreliable. RFC standards, which might help make the system more robust, are widely ignored. There is a large and growing discrepancy between our ability as a community to collect information and our ability to make this information useful for those under threat.

It is not clear where to go from here. One could find a bit of solace in the fact that network operators did much better in

terms of being reachable. Should we direct our notifications more to them? This will surely overload them. Their IP address space may be filled with hundreds, thousands, or even tens of thousands of affected systems. Another disadvantage is that in terms of remediation, this path was not more effective. Perhaps they are too far removed from the resource owner or operator to really do anything, except forward the notification. This is already a non-trivial task, which requires dynamically mapping notifications to the relevant customer in their network.

What else could be done? One option is to move away from email as the main notification medium. There are other options that are likely to be more effective, such as automated feeds, APIs, or sharing data within specific communities. For instance, Kührer et al. issued notifications (about systems vulnerable to abuse in NTP DDoS amplification attacks) to key organizations such as abuse team contacts at CERTs, security data clearinghouses [12].This indirect approach proved very effective: 92% of the amplifiers were remediated in three months.

The problem with these alternative information sharing mechanisms is that they are typically based on opt-in. Given that many of the affected parties in our studies didn't even set up a correct SOA record or put a working email address in their WHOIS record, it is difficult to be optimistic about any information sharing mechanism that requires an active effort on the side of the recipient. This question will have to be picked up by the industry, CERT and CSIRT community, Regional Internet Registries and others. Getting perfect reachability is unlikely to happen any time soon, but it should definitely be possible to improve beyond the current sorry state of affairs.

B. Incentives for remediation

While notifications did lead to more remediation than in the control groups, the overall remediation rates were low. Now, one issue is that not all vulnerabilities need be remediated. This fact is under-appreciated by the well-meaning efforts to increase vulnerability scanning and notification. Remediation represents an economic tradeoff and the outcome depends on the threat model of the affected party. This issue is undoubtedly also in play among the recipients of our notification. That being said, to offer total control over your DNS records to anyone on the Internet seems like an obvious problem that should be fixed. Some potential fixes, or perhaps it is better to call them workarounds, can be applied in a relatively simple manner. So why aren't they? Is it a lack of awareness? Incompetence? Lack of resources? The truth is: we don't know.

Security economics has taught us that systems are particularly prone to failure when the actor protecting it does not suffer the full cost of failure. Perhaps the incentive of the nameserver operator is too weak, as the abuse would impact the domain owner first and foremost. For this reason, we investigated if the incentive structure for remediation was stronger when we contacted the domain owners, who could then request the remediation from their provider, leveraging the commercial incentive of the latter party.

Our study found that this mechanism does not lead to better remediation. If remediation is a matter of incentives, then this indirect path either has equally weak incentives, or the stronger incentives are neutralized by the higher friction in the process towards remediation. In any case, the conclusion is that we need to look for other ways to improve the incentives. Some have pointed to reputation effects – a.k.a. naming, praising and shaming – as potentially effective [21].

C. Usefulness of the Demonstration Website

Another part of the incentive puzzle is more behavioral in nature. Recipients often need to triage notifications, and this will only increase in the age of large-scale vulnerability scanning. In this process, being able to asses the credibility, trustworthiness and criticality of the issue, might nudge recipients towards action. We tested whether mitigation improved when a website was provided with a live demonstration of the vulnerability for the recipient's domain or nameserver. The short answer is: no, remediation did not improve. The handful of responses to our survey do suggest, however, that the demonstration was helpful. So the bottleneck appears to be to get recipients to actually visit the site via a notification message. This is a complicated issue, as it triggers all kinds of overtones of phishing and other red flags for security-conscious persons. One way forward might be to host such a site at a trusted node in the network, such as the national CERT. Future work will have to test whether this has a more observable impact.

ACKNOWLEDGMENT

This publication was supported by a grant from the Netherlands Organisation for Scientific Research (NWO), under project number 628.001.022. Also, we would like to thank the anonymous reviewers, Andre Herdeiro Teixeira and Samaneh Tajalizadehkhoob for their helpful comments.

REFERENCES

[1] Orcun Cetin, Mohammad Hanif Jhaveri, Carlos Gañán, Michel van Eeten, and Tyler Moore. Understanding the role of sender reputation in abuse reporting and cleanup. *Journal of Cybersecurity*, 2(1):83–98, 2016.

[2] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami, Michael Bailey, Damon McCoy, Stefan Savage, and Vern Paxson. Youve got vulnerability: Exploring effective vulnerability notifications. In *USENIX Security Symposium (Aug. 2016)*, 2016.

[3] Maciej Korczyński, Michal Krol, and Michel van Eeten. Zone Poisoning: The How and Where of Non-Secure DNS Dynamic Updates. In *Proceedings of the 2016 ACM on Internet Measurement Conference*, pages 271–278. ACM, 2016.

[4] Paul Vixie, Susan Thomson, Yakov Rekhter, and Jim Bound. Dynamic Updates in the Domain Name System (DNS UPDATE). Internet RFC 2136, April 1997.

[5] D. Crocker. Mailbox Names for Common Services, Roles and Functions. RFC 2142 (Proposed Standard), May 1997.

[6] Abuse contact db. https://www.abusix.com/contactdb, note = Accessed: 2017-02-21.

[7] Serge Egelman, Lorrie Faith Cranor, and Jason Hong. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1065–1074. ACM, 2008.

[8] ICANN. Registrar accreditation agreement, 2013.
[9] A list of free email provider domains. https://gist.github.com/tbrianjones/5992856.
[10] Mohammad Hanif Jhaveri, Orcun Cetin, Carlos Gañán, Tyler Moore, and Michel Van Eeten. Abuse reporting and the fight against cybercrime. *ACM Computing Surveys (CSUR)*, 49(4):68, 2017.
[11] Ben Stock, Giancarlo Pellegrino, Christian Rossow, Martin Johns, and Michael Backes. Hey, you have a problem: On the feasibility of large-scale web vulnerability notification. In *USENIX Security Symposium (Aug. 2016)*, 2016.
[12] Marc Kührer, Thomas Hupperich, Christian Rossow, and Thorsten Holz. Exit from Hell? Reducing the Impact of Amplification DDoS Attacks. In *USENIX Security Symposium*, 2014.
[13] Zakir Durumeric, James Kasten, David Adrian, J Alex Halderman, Michael Bailey, Frank Li, Nicolas Weaver, Johanna Amann, Jethro Beekman, Mathias Payer, et al. The matter of heartbleed. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 475–488. ACM, 2014.
[14] Frank Li, Grant Ho, Eric Kuan, Yuan Niu, Lucas Ballard, Kurt Thomas, Elie Bursztein, and Vern Paxson. Remedying Web Hijacking: Notification Effectiveness and Webmaster Comprehension. In *Proceedings of the 25th International Conference on World Wide Web*, pages 1009–1019. International World Wide Web Conferences Steering Committee, 2016.
[15] Marie Vasek and Tyler Moore. Do Malware Reports Expedite Cleanup? An Experimental Study. In *CSET*, 2012.
[16] Antonio Nappa, M. Zubair Rafique, and Juan Caballero. Driving in the Cloud: An Analysis of Drive-by Download Operations and Abuse Reporting. In *Proceedings of the 10th Conference on Detection of Intrusions and Malware & Vulnerability Assessment*, pages 1–20, Berlin, Germany, July 2013. Springer.
[17] Davide Canali, Davide Balzarotti, and Aurélien Francillon. The role of web hosting providers in detecting compromised websites. In *Proceedings of the 22nd international conference on World Wide Web*, pages 177–188. International World Wide Web Conferences Steering Committee, 2013.
[18] Carlos Gañán, Orcun Cetin, and Michel van Eeten. An Empirical Analysis of ZeuS C&C Lifetime. In *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, pages 97–108. ACM, 2015.
[19] Marie Vasek, Matthew Weeden, and Tyler Moore. Measuring the impact of sharing abuse data with web hosting providers. In *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security*, pages 71–80. ACM, 2016.
[20] Qian Tang, Leigh Linden, John S Quarterman, and Andrew B Whinston. Improving internet security through social information and social comparison: A field quasi-experiment. *WEIS 2013*, 2013.
[21] Shu He, Gene Moo Lee, Sukjin Han, and Andrew B Whinston. How would information disclosure influence organizations outbound spam volume? evidence from a field experiment. *Journal of Cybersecurity*, 2016.

APPENDIX A
SAMPLE NOTIFICATION EMAILS

Label: Conventional Notification Content for Network Operators and Nameserver Operators
Subject: Vulnerable DNS Nameserver at ns1.example.com

Body: Cybersecurity researchers from Delft University of Technology have been conducting scans to identify DNS nameservers that are vulnerable to an attack called zone poisoning. The vulnerability allows an attacker to replace, add and remove Resource Records in authoritative zone files on the nameserver. In practice, this means an attacker can point the domain name to an IP address under the attackers control, add subdomains, or point existing subdomains, such as for email or ssh, to other IP addresses.

We scanned for this vulnerability by sending a single RFC-compliant DNS packet to all publicly visible nameservers. The response of your name server indicated that it is vulnerable to malicious dynamic updates. We did not exploit the server or interact with the existing records on it.

We have observed the following vulnerable nameservers on your network:

```
ns1.example.com
ns2.example.com
```

What can you do about this problem? The vulnerability can be mitigated by using an access control list on your name server, though this can still be circumvented via IP spoofing since the attack only needs a single UDP packet. The secure solution is to either disable so-called dynamic updates or to enable Transaction Signatures (TSIG) on the server and permitting only DNS dynamic updates with authorized keys.

Did you find this notification useful? Or do you object to these kinds of scans? We are doing research to make vulnerability and abuse notifications more effective for network operators and domain owners. Please help us to make them better for everyone by taking a 5 minute anonymous survey at: `http://www.surveylink.com/[surveylink]`

You can leaves us feedback via the survey or contact us directly at `vulnerabilityreporter@tudelft.nl`.

Thank you!

TU Delft Security Notifications Project

****

List of vulnerable domains:

```
example1.com
example2.com
example3.com
```

Label: Demonstrative Notification Content for Network Operators and Nameserver Operators
Subject: Vulnerable DNS Nameserver at ns1.example.com

Body: Cybersecurity researchers from Delft University of Technology have been conducting scans to identify DNS nameservers that are vulnerable to an attack called zone poisoning. The vulnerability allows an attacker to replace, add and remove Resource Records in authoritative zone files on the nameserver. In practice, this means an attacker can point the domain name to an IP address under the attackers control, add subdomains, or point existing subdomains, such as for email or ssh, to other IP addresses.

We scanned for this vulnerability by sending a single RFC-compliant DNS packet to all publicly visible nameservers. The response of your name server indicated that it is vulnerable to malicious dynamic updates. We did not exploit the server or interact with the existing records on it.

We have observed the following vulnerable nameservers on your network:

`ns1.example.com`

You can safely and easily test the vulnerability of your name server on our website at `zonepoisoning.com`. To prevent others from using the tool to search for vulnerable nameservers, we provide you with a unique token. Please use this URL to test domains using your nameserver(s):

`http://zonepoisoning.com/[uniquecode]`

You can use any of the vulnerable domain names mentioned at the bottom of this email to test the vulnerability, for example: example.com

Our website provides a simple interface that lets you add an innocent resource record to your nameserver for the subdomain zonepoisoning – for example `zonepoisoning.example1.com`.us. If the benign subdomain is successfully added, it means your server is vulnerable and all existing records can be changed from anywhere on the Internet! You can also use our website to check whether the vulnerability has been fixed.

What can you do to fix this problem? The vulnerability can be mitigated by using an access control list on your name server, though this can still be circumvented via IP spoofing since the attack only needs a single UDP packet. The secure solution is to either disable so-called dynamic updates or to enable Transaction Signatures (TSIG) on the server and permitting only DNS dynamic updates with authorized keys.

Did you find this notification useful? Or do you object to these kinds of scans? We are doing research to make vulnerability and abuse notifications more effective for network operators and domain owners. Please help us to make them better for everyone by taking a 5 minute anonymous survey at: `http://www.surveylink.com/[surveylink]`

You can leaves us feedback via the survey or contact us directly at vulnerabilityreporter@tudelft.nl.

Thank you!

TU Delft Security Notifications Project

****

List of vulnerable domains:

```
example1.com
example2.com
example3.com
```

Label: Conventional Notification Content for Domain Owners
Subject: Vulnerable Nameserver for example.com

Body: Cybersecurity researchers from Delft University of Technology have been conducting scans to identify domain names that are vulnerable to an attack called zone poisoning. The vulnerability allows an attacker to replace, add and remove Resource Records in authoritative zone files on the nameserver for that domain. This is a critical security risk.

An attacker can easily point your domain name to an IP address under the attackers control, add subdomains, or point existing subdomains, such as for email or ssh, to other IP addresses.

We scanned for this vulnerability by sending a single RFC-compliant DNS packet to all publicly visible nameservers. We did not exploit the nameserver or interact with the existing records on it.

We found that the following domain is vulnerable to this attack:

`example.com`

The vulnerability can be mitigated by changing the configuration of the authoritative nameserver. If your domain is hosted at a hosting provider, you might not have any control over the nameserver. In that case you need to contact the hosting provider or whoever operates the nameserver for your domain.

One way to mitigate the vulnerability is to use an access control list on the nameserver, though this can still be circumvented via IP spoofing since the attack only needs a single UDP packet. The secure solution is to either disable so-called dynamic updates or to enable Transaction Signatures (TSIG) on the server and permitting only DNS dynamic updates with authorized keys.

Did you find this notification useful? Or do you object to these kinds of scans? We are doing research to make vulnerability and abuse notifications more effective for domain owners and network operators. Please help us to make them better for everyone by taking a 5 minute anonymous survey at: `http://www.surveylink.com/[surveylink]`

You can leaves us feedback via the survey or contact us directly at: vulnerabilityreporter@tudelft.nl.

Thank you!

TU Delft Security Notifications Project

****

List of vulnerable nameserver(s) associated with your domain(s):

```
ns1.example2.com
ns2.example3.com
```

Label: Demonstrative Notification Content for Domain Owners
Subject: Vulnerable Nameserver for example.com

Body: Cybersecurity researchers from Delft University of Technology have been conducting scans to identify domain names that are vulnerable to an attack called zone poisoning. The vulnerability allows an attacker to replace, add and remove Resource Records in authoritative zone files on the nameserver for that domain. This is a critical security risk. An attacker can easily point your domain name to an IP address under the attackers control, add subdomains, or point existing subdomains, such as for email or ssh, to other IP addresses.

We scanned for this vulnerability by sending a single RFC-compliant DNS packet to all publicly visible nameservers. We did not exploit the nameserver or interact with the existing records on it.

We found that the following domain is vulnerable to this attack:

`example.com`

You can safely and easily test the vulnerability of these domains on our website `zonepoisoning.com`. To prevent others from using the tool to search for vulnerable domains, we provide you with a unique token. Please use this link and the domain(s) listed above to test whether your domain has the vulnerability: `http://zonepoisoning.com/[uniquecode]`

On our website, we provide a simple interface that lets you add an innocent subdomain to your domain: zonepoisoning.baekryongdoosio.com. If the subdomain is successfully added, it means the nameserver is vulnerable and all DNS records associated with your domain can be changed from anywhere on the Internet! You can also use our website to check whether the vulnerability has been fixed.

The vulnerability can be mitigated by changing the configuration of the authoritative nameserver. If your domain is hosted at a hosting provider, you might not have any control over the nameserver. In that case you need to contact the hosting provider or whoever operates the nameserver for your domain.

One way to mitigate the vulnerability is to use an access control list on the nameserver, though this can still be circumvented via IP spoofing since the attack only needs a single UDP packet. The secure solution is to either disable so-called dynamic updates or to enable Transaction Signatures (TSIG) on the server and permitting only DNS dynamic updates with authorized keys.

Did you find this notification useful? Or do you object to these kinds of scans? We are doing research to make vulnerability and abuse notifications more effective for domain owners and network operators. Please help us to make them better for everyone by taking a 5 minute anonymous survey at: `http://www.surveylink.com/[surveylink]`

You can leaves us feedback via the survey or contact us directly at: vulnerabilityreporter@tudelft.nl.

Thank you!

TU Delft Security Notifications Project

****

List of vulnerable nameservers associated with your domain(s):

```
ns1.example2.com
ns2.example3.com
```

APPENDIX B
SURVEY QUESTIONNAIRE

Security Notification Survey

Please help us improve security notifications by answering a 2-minute anonymous survey. Each question is optional, please answer the ones that you feel comfortable with. Your feedback is very important to us and we really appreciate your time.

Common Questions

1) Did your organization take prior actions to resolve the security issue before our notification?
   a) Yes
   b) No
2) Is your organization planning on resolving the security issue?
   a) Yes
   b) No
3) Do you feel it was acceptable for us to scan the nameserver for this security issue?
   a) Yes
   b) No
4) Do you feel it was acceptable for us to notify your organization?
   a) Yes
   b) No
5) Would your organization want to receive similar security vulnerability/misconfiguration notifications in the future?
   a) Yes
   b) No
6) Did we notify the correct contact?
   a) Yes
   b) No
7) Is there anything you want to tell us about our scans, notifications or any other issue related to this research or to security notifications in general?

Specific Questions to Nameserver Operators and Network Operators

1) How would you characterize your organization?
   a) Hosting provider
   b) Reseller
   c) DNS server provider (Only in 1st Campaign)
   d) ISP broadband
   e) Content delivery/distribution network
   f) Registrar
   g) Other - Write In ...
2) How many employees work at your organization?
   a) 1
   b) 2-24
   c) 25-99
   d) 100-499
   e) 500-999
   f) 1,000+
   g) Other - Write In ...

Specific Questions to Domain Owners

1) You are the contact for this domain. How would you characterize yourself?
   a) I am an individual who owns this domain
   b) I am a member of the organization who owns this domain
   c) Other - Write In ...
2) If the domain is owned by an organization, how large is this organization?
   a) 1-24
   b) 25-99
   c) 100-499

d) 500-999
e) 1,000+
f) Other - Write In ...

Specific Questions for Network Operators and Domain Owners

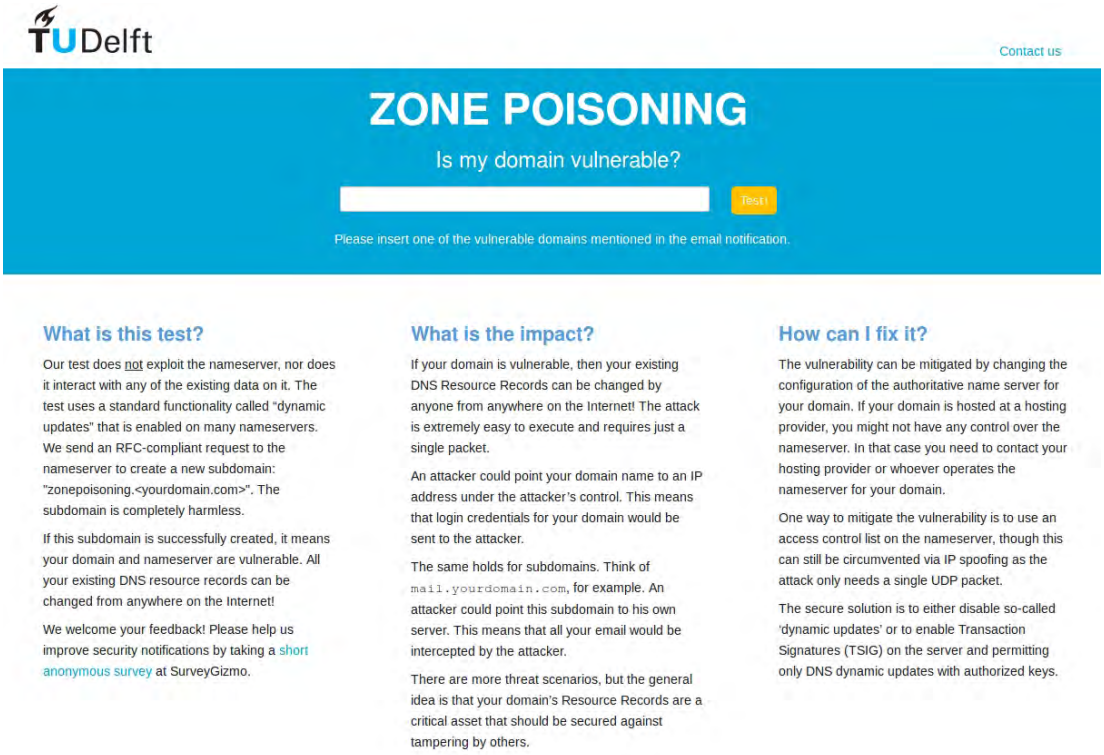
1) Is your organization in charge of maintaining name server?
   a) Yes
   b) No

Specific Questions for the Demo Group

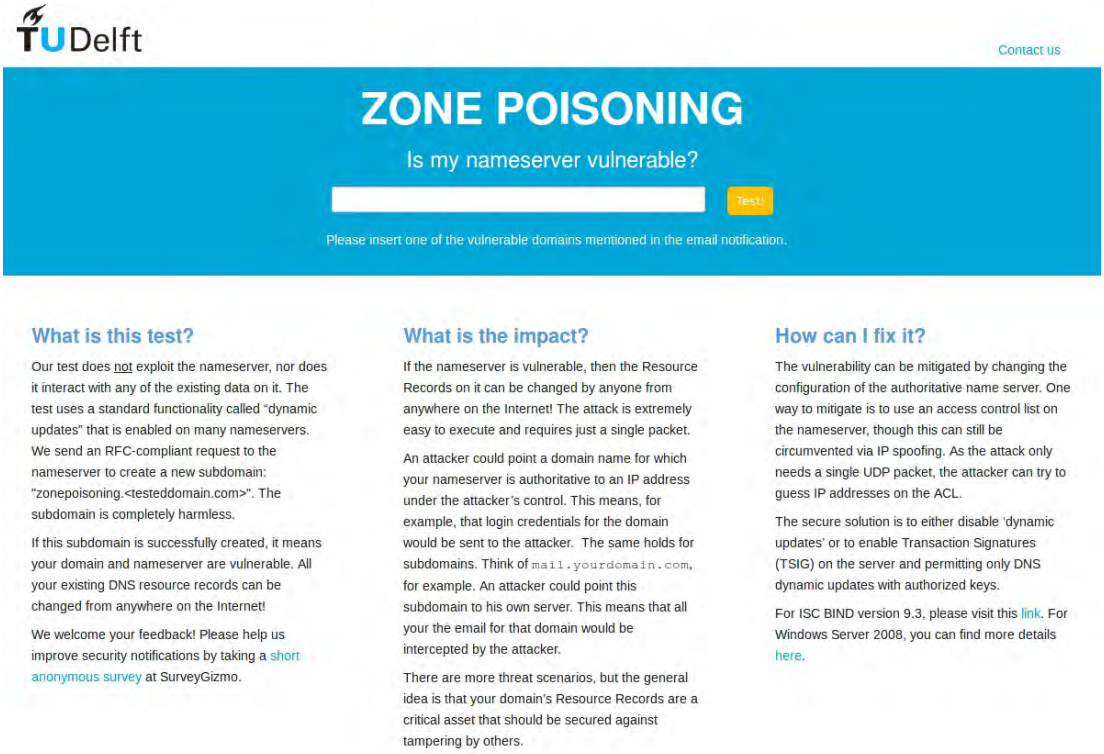
1) We set up the site zonepoisoning.com to enable you to safely demonstrate the security issue. Do you feel this was useful?
   a) I went to the site, but I did not find it useful
   b) I found it somewhat useful
   c) I found it very useful
   d) I did not go to the site
   e) Other - Write In ...

Specific Questions for the Conventional Notification Group

1) Would it have been useful if we had provided you with a site where you could safely test and demonstrate the security issue?
   a) Not useful
   b) Somewhat useful
   c) Very useful
   d) Dont know
   e) Other - Write In ...

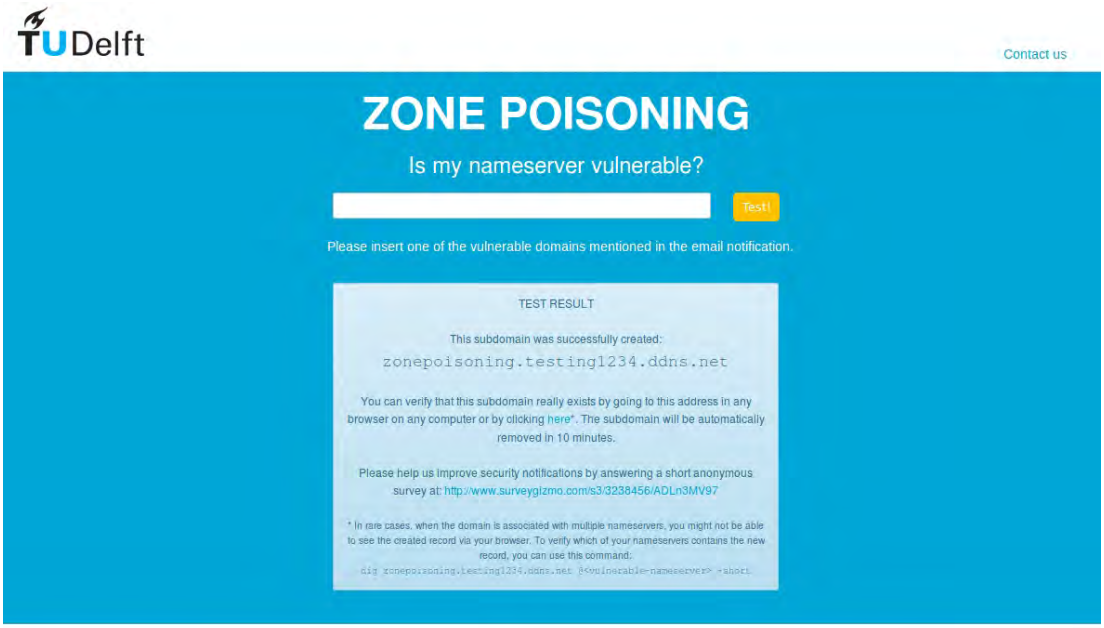


(a) Demo. website for domain owners

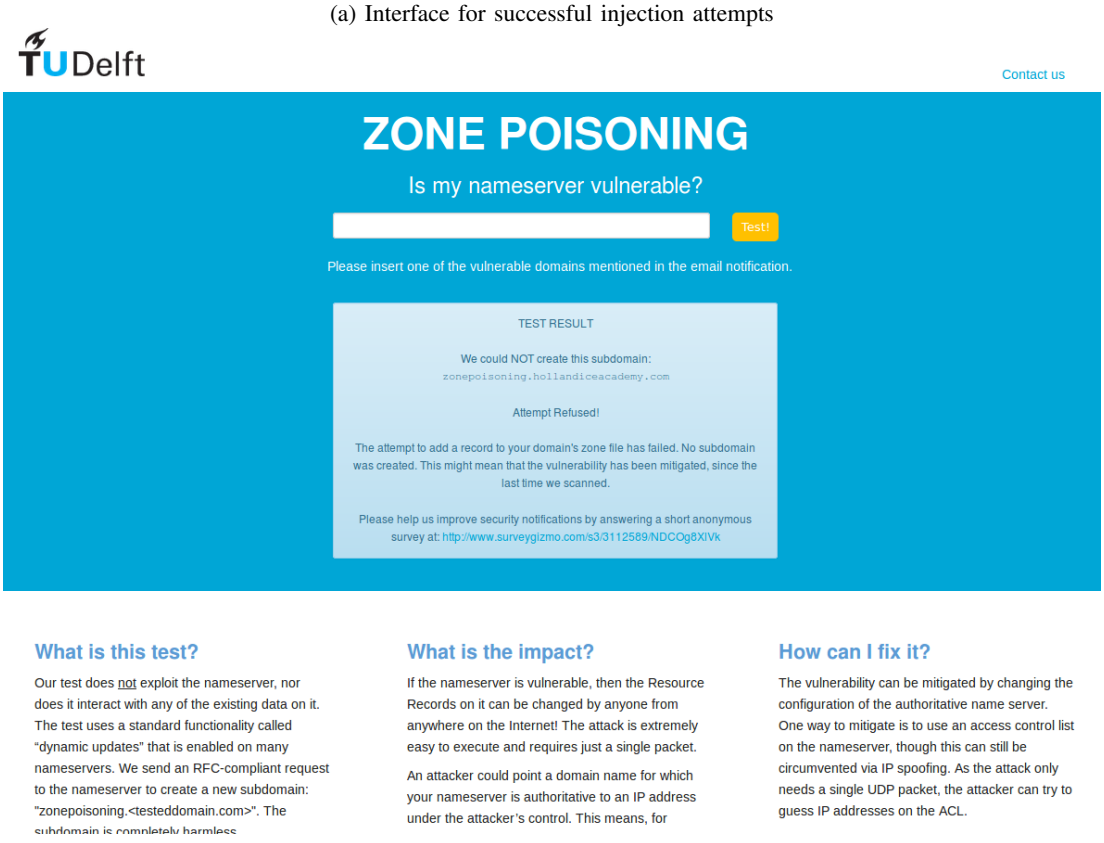


(b) Demo website for nameserver operators

Fig. 7: Vulnerability demonstration website



(a) Interface for successful injection attempts



(b) Interface for unsuccessful injection attempts

Fig. 8: Website interface after user interaction

Welcome to the "DNS dynamic update measurement" server project

Some questions you may have:

1. What computer is this?
   - This is a computer from the Economics of CyberSecurity group from Delft University of Technology.
   - It is run by researchers from this group.
2. Why are we sending the DNS dynamic update packets to your server?
   - We carry out Internet-wide measurements in order to track the population of servers vulnerable to the nonsecure DNS dynamic updates and notify the system operators responsible for misconfigured machines.
3. Where can I learn more about secure DNS dynamic updates?
   - Please visit this website (ISC BIND version 9.3) or this website (Windows Server 2008) for more details.
4. What update do you send to my DNS server?
   - The following `A` record: `researchdelft.2ndLevelDomainName 86400 A 192.42.131.1`
5. OK, if it is for research, I have no problem with it.
   - Thanks so much! We appreciate it.
6. No, I want you to stop sending the DNS dynamic update packets to my server.
   - Please just write to us (maciej [dot] korczynski [a_t] tudelft [dot] nl) (PGP: 848571D0) and we will include your zone/name server in our not-scan list.

Fig. 9: Destination of injected record