Don't Get Hijacked: Prevalence, Mitigation, and Impact of Non-Secure DNS Dynamic Updates

Yevheniya Nosyk, Maciej Korczyński, Carlos H Gañán, Michal Król, Qasim Lone, Andrzej Duda

▶ **To cite this version:**

Yevheniya Nosyk, Maciej Korczyński, Carlos H Gañán, Michal Król, Qasim Lone, et al.. Don't Get Hijacked: Prevalence, Mitigation, and Impact of Non-Secure DNS Dynamic Updates. IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2023), Nov 2023, Exeter, United Kingdom. pp.1480-1489, 10.1109/TrustCom60117.2023.00202 . hal-04602230

HAL Id: hal-04602230
https://hal.science/hal-04602230v1

Submitted on 5 Jun 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Don't Get Hijacked: Prevalence, Mitigation, and Impact of Non-Secure DNS Dynamic Updates

Yevheniya Nosyk*, Maciej Korczyński*,Carlos H. Gañán‡, Michał Król§, Qasim Lone¶, Andrzej Duda*
*Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, France
Email: firstname.lastname@univ-grenoble-alpes.fr
‡TU Delft, The Netherlands
§City, University of London, The United Kingdom
¶RIPE NCC, The Netherlands

***Abstract*—DNS dynamic updates represent an inherently vulnerable mechanism deliberately granting the potential for any host to dynamically modify DNS zone files. Consequently, this feature exposes domains to various security risks such as domain hijacking, compromise of domain control validation, and man-in-the-middle attacks. Originally devised without the implementation of authentication mechanisms, non-secure DNS updates were widely adopted in DNS software, subsequently leaving domains susceptible to a novel form of attack termed zone poisoning. In order to gauge the extent of this issue, our analysis encompassed over 353 million domain names, revealing the presence of 381,965 domains that openly accepted unsolicited DNS updates. We then undertook a comprehensive three-phase campaign involving the notification of Computer Security Incident Response Teams (CSIRTs). Following extensive discussions spanning six months, we observed substantial remediation, with nearly 54% of nameservers and 98% of vulnerable domains addressing the issue. This outcome serves as evidence that engaging with CSIRTs can prove to be an effective approach for reporting security vulnerabilities. Moreover, our notifications had a lasting impact, as evidenced by the sustained low prevalence of vulnerable domains.**

***Index Terms*—DNS, dynamic updates, notifications.**

I. Introduction

In the early stages of the Internet, hosts were primarily identified by their IP addresses, which were difficult to remember. To alleviate this issue, the Stanford Research Institute introduced the static `HOSTS.TXT` file, which facilitated the mapping of host names to IP addresses. As the number of network-connected devices rapidly increased, a more scalable solution was required. The Domain Name System (DNS), standardized in 1987 [32], [33] met the requirement.

The early DNS was relatively static. Domain owners occasionally updated local copies of zone files, but the whole process was not automated and could not scale. However, with the emergence of the Dynamic Host Configuration Protocol (DHCP) [13], it became essential to promptly assign domain names to dynamically added hosts. In 1997, the IETF published a new RFC 2136 [51] proposed standard called "Dynamic Updates in the Domain Name System (DNS UPDATE)." This new mechanism allowed dynamically updating the content of zone files of authoritative nameservers. Notably, the authors stated that unless coupled with some external security mechanism, any host on the Internet would be able to update external zone files by sending a single UDP packet. As DNS was becoming increasingly ubiquitous and thus an attractive attack target, such protocol extensions became particularly dangerous for domain owners.

DNS has long been an attractive target for attackers. Domain shadowing [30], cache poisoning attacks [43], and other forms of DNS manipulation [36], [37] may remain unnoticeable for a while. Non-secure DNS updates make such attacks trivial—they eliminate the need to steal any credentials and allow modifying target zone files directly. Contrary to cache poisoning attacks that affect individual recursive resolvers, modified zone files are globally accessible. Adopting the terminology of the previous work [23], we refer to the attacks exploiting non-secure DNS dynamic updates as *zone poisoning*.

While zone poisoning attacks can have devastating consequences for domain owners, they received very little attention. To the best of our knowledge, only one paper enumerated domain names vulnerable to zone poisoning in the wild. In 2016, researchers scanned a sample of 2.9M randomly chosen domains and 1M domains from the Alexa popularity list. Although that study gives the initial glance at the prevalence of the vulnerability in the wild, the coverage is low, and previous attempts to mitigate the problem did not result in substantial remediation [7]. In this paper, we extend the work of Korczyński et al. [23] and present the following contributions:

1) **We define an extensive zone poisoning attack taxonomy.** We present five classes of attacks that can be performed on domains supporting non-secure DNS dynamic updates. We experimentally verify those attacks in a controlled test environment.
2) **We scan more than 353M domain names and 3.6M nameservers—two orders of magnitude more than previous work.** We test whether they accept non-secure DNS dynamic updates from arbitrary clients and identify 200 times more vulnerable domains (382K) and 5 times more (5.6K) vulnerable nameservers. We analyze their distribution across autonomous systems and countries.
3) **We lead a large-scale notification campaign to fix vulnerable resources.** We send carefully-crafted notification emails containing various nudges to CSIRTs and actively engage in discussion with them. We show that contacting CSIRTs can be an efficient way to report vulnerabilities

and have them fixed (contrary to the common disbelief). As a result, almost 54% of nameservers and 98% of vulnerable domain names were remediated.

4) **After carefully analyzing the attack, we followed a responsible disclosure procedure.** At the time of writing, two CVEs have been reserved for the impacted DNS vendor software: Knot DNS and Simple DNS Plus.

II. Background

This section introduces the necessary background on dynamic updates in DNS, the associated security risks, and implementations in popular DNS software.

A. Dynamic Updates in DNS

Dynamic Updates in DNS were introduced in the proposed standard RFC 2136 [51] back in 1997 to address the need to update the content of DNS zones dynamically. They allowed efficient updating, adding, or deleting any type of resource record (RR), e.g., `A`, `AAAA`, `NS`, etc.

If the authoritative nameserver supports dynamic updates, it inspects the packet to identify whether all update prerequisites (if any) are met and whether the client is allowed to request such an update. Note that unless restricted by the nameserver, anyone who knows the zone name (e.g., `example.com`) and the nameserver (e.g., `ns1.example.com`) is capable of updating the zone content by sending a single UDP datagram.

B. Security Considerations

Dynamic updates raise a vital concern—if configured insecurely, they will be accepted from any host on the Internet. As acknowledged in RFC 2136, non-secure updates constitute "a serious increase in vulnerability from the current technology." The subsequent RFC 2137 [1] and RFC 3007 [53] proposed using cryptographic keys to generate signatures covering update requests. This would only allow authorized clients to perform updates but adds more complexity related to key management. As an alternative, a lightweight security mechanism based on shared secret keys and one-way hashing was introduced three years later in RFC 2845 (TSIG: Secret Key Transaction Authentication for DNS) [2].

If none of the cryptographic security mechanisms is implemented, an authoritative nameserver is expected only to accept the dynamic updates from a statically preconfigured set of IP addresses. The address match lists should be as restrictive as possible and limited to, for example, an IP address of a DHCP server. Nevertheless, such a mechanism is still insecure because the adversary may guess the IP addresses from the list and send requests with spoofed source IPs. Such access control lists could have been effective if dynamic updates were using the TCP protocol for transport. However, Paul Vixie–the editor of RFC 2136–explained to us that the working group was considering proposing TCP rather than UDP, but that would involve opening the port TCP 53 in firewalls, which seemed to be problematic at that time. Therefore, they decided to release the document as a *proposed standard* instead of an *Internet standard* at least until the security issues raised in the specification are not addressed.

C. Implementation of Dynamic Updates

We analyzed five popular implementations of DNS authoritative nameserver software and verified whether they support dynamic updates and what default settings are offered to clients as of April 2023:

- **BIND 9.18.4:** by default, dynamic updates are disabled and must be explicitly configured using either `allow-update` or `update-policy` options. The first statement (`allow-update`) is a simple access control list (ACL) that grants permission to update the zone for any address matching the list. This option, however, raises several security issues. First, an administrator may use a built-in `any` argument to accept updates from all IP ranges. Second, even if the ACL is restricted to individual IP addresses, an attacker may send `UPDATE` requests with spoofed source IP addresses from the ACL. The official BIND9 manual [19] strongly recommends avoiding IP-based ACLs and specifying `TSIG` key names instead. The second statement (`update-policy`) is restrictive as it only allows `TSIG`-based access lists and does not let one specify a range of IP addresses.
- **Windows Server 2022:** distinguishes between two types of zones: standard primary and directory-integrated. The latter support either "secure only" updates using extended `TSIG` or "nonsecure and secure" updates–a zone type that accepts updates from any client. Standard primary zone configuration supports *only* "nonsecure and secure" updates. Microsoft is aware of the vulnerability and informs the users willing to set up a non-secure implementation that "allowing nonsecure dynamic updates is a significant security vulnerability because updates can be accepted from untrusted sources."
- **PowerDNS 4.6.2:** dynamic updates are implemented but disabled by default. They are explicitly activated in the main configuration file with the `dnsupdate=yes` statement. Updates are only accepted from IP address ranges defined under `allow-dnsupdate-from` (the default is `127.0.0.0/8`, but `0.0.0.0/0` would match any IPv4 address) [39]. If the secondary server receives the update, it is automatically forwarded to the primary server (provided all the permissions are granted). More options can be configured per each individual zone, for example, the list of allowed addresses or `TSIG`.
- **Knot DNS 3.1.9:** Dynamic updates are implemented but need to be explicitly enabled with the `acl` statement and added to each corresponding `zone` statement [11]. By default, the `address` field in `acl` will match any source IP address unless a more precise range is provided. This is a serious security threat that allows any host on the Internet to send dynamic updates. Nevertheless, secure updates are also available and can be configured by adding a `key` statement and then referring to it inside the `acl`. Similarly to PowerDNS or BIND9, updates received by the secondary server are forwarded to the primary server.
- **Simple DNS Plus 9.1:** this software implements standard

(non-secure) and TSIG-signed dynamic updates, both disabled by default [20]. When configuring standard updates, the administrators can either accept them from any IP address or create an ACL of address ranges.

III. Adversary Model

In this section, we provide a taxonomy of attacks conducted by an adversary on a vulnerable authoritative nameserver. We begin by describing our experimental setup and proceed to outline the attack vectors tested in our laboratory environment.

A. Infrastructure Setup

To define the adversary model and validate it in a controlled environment, we established our infrastructure consisting of two servers:

- **The adversary:** We assume that our adversary has already conducted scans for vulnerable resources and possesses knowledge of the domain name and its authoritative nameserver. The adversary also requires the use of the standard DNS dynamic update utility `nsupdate` [29] or a dedicated software for modifying the victim's zone. For more sophisticated attacks, the adversary may need to configure a mail, web, or DNS server.
- **The victim:** We configure the `example.com` test zone and the `ns1.example.com` nameserver using BIND9 software. It is configured to accept non-secure dynamic updates from any host on the Internet, as governed by the `allow-update` option. Furthermore, we host a website using the Apache HTTP server software, deploy SSL/TLS certificates, and operate a mail server (`mail.example.com`) with the corresponding `MX` record defined in the zone file.

B. Taxonomy of Attacks

Below are five categories of zone poisoning attacks, each with its own trade-off between viability and stealthiness.

1) Denial of Service (DoS) Attack: This category of attacks is relatively simple to execute, although they lack significant stealthiness, as the victim would promptly detect the unresponsive domain and take measures to rectify the configuration.

- **Deletion of an `A/AAAA` record:** this removes the mapping between the domain name (`example.com`) and its corresponding IP address. Subdomains can be completely removed if managed by the same parent nameserver (`ns1.example.com`). If the victim operates different services, such as accounts, mail, FTP, or checkout, on separate subdomains, the adversary may selectively disable specific services, making detection less straightforward.
- **Deletion of an `MX` record:** similarly, the adversary can also delete the mail exchange record (and/or its glue record) that specifies the name of the mail server that accepts email messages on behalf of the domain name. It will not only disrupt the email service itself but also hinder any abuse (notification) messages sent to the victim.
- **Deletion, addition, or modification of a `TXT` record:** `TXT` records are widely used for DNS-based service discovery, Domain-based Message Authentication, Domain Keys Identified Mail, DMARC, SPF, and more [9], [22], [24], [25]. For example, an attacker can modify an existing `TXT` SPF record to `v=spf1 -all`, blocking all hosts from sending emails on behalf of `example.com`. While this attack is highly feasible, its stealthiness diminishes due to prompt detection of the manipulated `TXT` record.

2) Domain Hijacking Attack: Modifies original DNS records so that the legitimate traffic is redirected to bogus servers under the attacker's control.

- **Update of an `A/AAAA` record:** an adversary can update the victim's `A/AAAA` record and replace the legitimate IP address with the one under the attacker's control. As a result, all the client traffic (e.g., website visitors) will be diverted from the domain owner to the adversary.
- **Update of an `MX` record:** if the victim operates a mail server, the adversary can access all email traffic associated with the victim's domain. They can achieve this by setting up a malicious mail server, such as `mail.malicious.com`, and modifying the `MX` record. If the vulnerable nameserver manages the domain, the adversary can update the `A/AAAA` record of `mail.example.com` with their own IP.

3) Man-in-the-Middle (MITM) Attack: This advanced attack requires greater sophistication from the adversary, making it highly stealthy and difficult to detect. The adversary not only manipulates DNS records like in domain hijacking attacks but also intercepts and redirects the traffic to the victim's server.

- **Update of an `A/AAAA` record:** the adversary may establish a proxy between the victim and its clients to either passively observe the traffic or create a malicious service to exploit the victim's customers and extract sensitive information like login credentials. In both cases, the adversary initiates the attack by modifying an `A/AAAA` record of a domain or subdomain.
- **Update of an `MX` record:** similarly to domain hijacking, the adversary must update the `MX` resource record to redirect all traffic to the malicious mail server. However, in this case, the mail traffic will be further forwarded to the intended recipients on behalf of the hijacked victim.

4) Domain Shadowing Attack: Involves creating malicious subdomains for exploit kits, malware, phishing, and client information theft. These attacks are highly stealthy as subdomains leverage the trust of the parent domain. The adversary can create multiple subdomains effortlessly and rotate between them to avoid detection.

- **Addition of an `A/AAAA` record:** create numerous malicious subdomains that direct to web servers hosting, for example, malware or phishing websites. This involves adding corresponding `A` or `AAAA` records to the zone file, such as assigning `1.2.3.4` to `paypal.account.example.com`. To enhance the attack's stealthiness and evade IP-based blocklisting, the adversary may employ the fast-flux technique, dynamically associating multiple subdomains with a wide range of IP addresses.

- **Addition of an `NS` record:** Generating multiple subdomains and rotating IP addresses helps evade blocklisting services, but frequent changes in the zone file may alert the victim. To enhance the stealthiness of the domain shadowing attack, the adversary can introduce a malicious delegation (`account.example.com`) and set up a nameserver (`ns1.account.example.com`). This allows the adversary to create subdomains (`paypal.account.example.com`) directly on the malicious nameserver, bypassing the victim's server.

5) Compromising Domain Control Validation: Digital certificates establish domain authenticity, verified through Domain Control Validation (DCV) to prove ownership. However, domain hijacking and other techniques discussed can bypass this validation. These stealthy attacks involve temporary zone file modifications during certificate issuance.

- **Update of an `A/AAAA` record:** HTTP-based validation requires uploading a file from the certification authority to a particular directory on the web server. The file must remain accessible over HTTP. For example, to obtain a certificate for `example.com`, a file must be uploaded accessible on `http://example.com/.well-known/pki-validation/filename.txt` to validate the ownership. If the adversary temporarily updates the web server's IP address (`A/AAAA` record), the certificate authority will look for a file on a malicious web server.
- **Addition of a `CNAME` record:** DNS-based validation may require domain owners to add a `CNAME` record to the zone file. The record contains a random string generated by the certificate authority, which requests the `CNAME` over DNS. If the adversary uploads such a record to the victim zone file, the domain ownership will be validated, and the attacker will obtain the digital certificate.

C. Adversary Capabilities: Additional Insights

This section explores the adversary's capabilities in zone poisoning attacks, specifically regarding the propagation of updates between primary and secondary nameservers and the vulnerability of IP-based access control lists to IP spoofing [31]. These insights highlight the need for secure dynamic updates and TSIG-based access lists to mitigate the risks associated with these attack vectors.

1) Propagation Between Primary and Secondary Nameservers: A single DNS zone may be served by multiple nameservers (usually a primary and a secondary), so we further investigate whether unsolicited DNS updates would propagate between the two. We set up a primary and a secondary nameservers, but only enable dynamic updated at one of those at a time. In the first case, only the primary nameserver received an update packet. We confirmed that the newly added resource record was immediately propagated to the secondary nameserver via the DNS Incremental Zone Transfer Protocol (IXFR) mechanism. In the second scenario, the secondary nameserver accepted non-secure dynamic updates from arbitrary clients while the primary allowed updated from the secondary only. Upon sending the update packet to the secondary nameserver, it did not update its zone file directly but rather forwarded the request to the primary nameserver. The primary, in turn, updated its zone file and initiated a zone transfer to the secondary, thus adding a new resource record to both nameservers.

Table I
TESTED AND VULNERABLE RESOURCES IDENTIFIED DURING THE GLOBAL AND SUBDOMAINS SCANNING CAMPAIGNS IN FEBRUARY-MARCH 2017

	Global Scan		**Subdomains Scan**	
	All tested	Vulnerable	All tested	Vulnerable
Domains	353,870,510	381,965 (0.108%)	35,382,217	399 (0.0011%)
NS IPs	3,855,615	5,575 (0.145%)	722,989	401 (0.0555%)
Domain–NS IPs	5,032,117,394	679,930 (0.014%)	104,955,041	520 (0.0005%)

2) Dynamic Updates with Spoofed Source IPs: In the aforementioned attack scenarios, we configured authoritative nameservers to freely accept non-secure updates. To prevent unsolicited zone changes, DNS administrators may create IP-based ACLs that would only allow authorized hosts (e.g., secondary nameservers, DHCP servers, machines from the same local network, etc.) to update zone files. Such configuration still remains highly insecure—as update packets are sent over UDP, an attacker can guess the authorized IP address and request a zone update on its behalf. We experimentally verified this attack scenario by configuring our nameserver to only accept updates from a particular IP address. We then sent an update packet with the spoofed source IP address from a different machine and confirmed that it was accepted by the nameserver. IP spoofing can, therefore, greatly increase the attack surface and target those nameservers that are seemingly secured and protected from arbitrary zone updates.

IV. ENUMERATION OF VULNERABLE RESOURCES

This section identifies the domains and the corresponding authoritative nameservers vulnerable to zone poisoning.

A. DNS Datasets

1) Global Scan: Sending a dynamic update requires two pieces of information: the zone name and the nameserver's IP address. Both can be found in passive DNS datasets or queried actively. We aggregated seven passive DNS datasets: i) Farsight's DNSDB [15], ii) Censys's Internet-Wide Scan Data Repository [40], iii) `.com`, `.net` and `.name` zone files provided by Verisign [50], iv) `.nl` zone file under the contract with SIDN–the `.nl` ccTLD registry [41], v) `AXFR` transfers of `.se` and `.nu` ccTLD [16], vi) `.us` ccTLD, `.biz`, `.org`, `.asia`, `.info`, `.mobi`, `.post` and `.tel` legacy gTLDs and 1230 new gTLDs made available by ICANN through the Centralized Zone Data Service [18], and vii) Alexa Top 1M [4].

From each dataset, we extracted all the second-level and upper-level domain names (if registered under public suffixes [34], e.g., `example.co.uk`). Next, we checked whether passive datasets contained nameserver records (`NS`) and the corresponding glue records (IPv4 addresses). If not

available, we actively queried the missing data. As shown in Table I, we gathered more than 353M unique domain names and 3.8M unique IPv4 addresses of nameservers, which render more than 5B domain name–nameserver pairs for scanning.

2) Subdomains Scan: In DNS, a registered domain may contain multiple subdomains, possibly served by different nameservers. In the case of such delegations, parent and child nameservers can have different DNS dynamic update policies. We extract all the domains with three or more labels (depending on the public suffix's length) from Farsight's [15] passive query traces. We later compare whether parent and child (subdomain) authoritative nameservers have consistent configurations for DNS dynamic updates. We additionally queried all the missing data (i.e., `NS` records and/or their corresponding IPv4 addresses). From 35M domain names and 722K nameservers, we created a scanning input list with approximately 105M entries (see Table I).

B. Scanning Methodology

We developed an efficient scanner capable of sending DNS update packets at scale. Each update attempts to insert a new `A` resource record to the zone file. Specifically, we add a new subdomain in format `researchstudyzp.example.com`, where `example.com` is the tested zone's name. The IP address is the one of our web server. It hosts a web page describing who we are, why we send DNS updates, how to correctly configure the server, and how to contact us.

To undoubtedly confirm a vulnerability, we perform an active DNS lookup to resolve the subdomain to our web server's IP address. If this succeeds, then the vulnerability is present. Finally, we remove the test DNS record by sending a delete request and then try to resolve it again to confirm the removal. We designed our scanning methodology so that only one UDP packet would be enough to test whether nameservers are vulnerable to the zone poisoning attack. We further discuss the ethical aspects of this study in Section VII.

C. Scan Results

We performed the global scan of domains and nameservers vulnerable to zone poisoning attack during four weeks in February-March 2017. Table I presents the proportion of vulnerable resources found in each category (domain names, nameserver IP addresses, and domain-nameserver IP address pairs)—orders of magnitude more than the previous work [23]. While the ratio remains low (less than 1% for each category), it translates into considerable absolute numbers. Overall, almost 382K domain names were found vulnerable to zone poisoning attacks. Given that those were often reachable over multiple vulnerable nameservers, 680K combinations of domain names and nameservers accepted arbitrary update requests.

To understand the population of vulnerable domains, we categorized them using Webshrinker [52]—a tool that uses artificial intelligence to automatically classify domains under the IAB taxonomy [3]. Such services are usually most effective when applied to domains that host websites, which is not necessarily the case in our input list. Nevertheless, we categorized

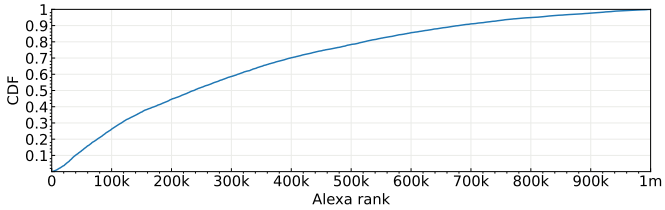

Figure 1. The distribution of 5,964 vulnerable domains in the Alexa domain popularity list.

1.5% of vulnerable domains. The "Non-Standard Content" is the most popular category representing message boards, content servers, or adult content (1089 domain names). The second most popular category is "Business", containing almost 800 domains, including banks and financial institutions. There is also a significant number of domains belonging to governmental (359), educational (483), and healthcare institutions (302). Therefore, some of the vulnerable domains represent critical services that are generally expected to be well-secured.

To further access the popularity of vulnerable domain names, we aggregate the Alexa 1M [4] top website lists in 2017. We found 5,964 vulnerable domains in the ranking (the most popular domain reaching the 244th place on the list) and plotted the distribution of ranks in Figure 1. Overall, vulnerable domains are evenly distributed across the popularity list.

We followed the global scan with a subdomains scan. The results are presented in Table I. The ratio of identified vulnerable resources is up to 3 orders of magnitude lower than for the global scan. This comes from the nature of our input list. Lengthy multiple-level domains are likely to be disposable (i.e., generated for one-time use) [8]. As a result, only 399 domains were vulnerable out of more than 35M and only 401 nameservers accepted non-secure updates for subdomains. Furthermore, we aggregated all the vulnerable subdomains by their corresponding second-level domains. Only 14 out of 236 aggregated second-level domains were vulnerable to zone poisoning. This finding highlights that subdomain servers may be vulnerable to our attack even when the delegating second-level domain servers are properly configured.

D. Unveiling Zone Poisoning Attacks

Zone poisoning attacks were previously unreported, prompting our investigation into their existence in the wild. Following the methodology outlined by Korczyński et al. [23], we utilized one year of passive DNS data provided by Farsight Security (2017). Our analysis focused on extracting queries related to subdomains of domains susceptible to zone poisoning, resulting in a total of 703K entries.

Our hypothesis is that if attackers were to exploit regular domains accepting non-secure updates from arbitrary clients, they might engage in domain shadowing, creating subdomains involved in malicious activities. These subdomains may then be reported and added to URL or domain blocklists. To validate our hypothesis, we compared our domain list with APWG [5] and Phishtank [10] feeds from the same period.

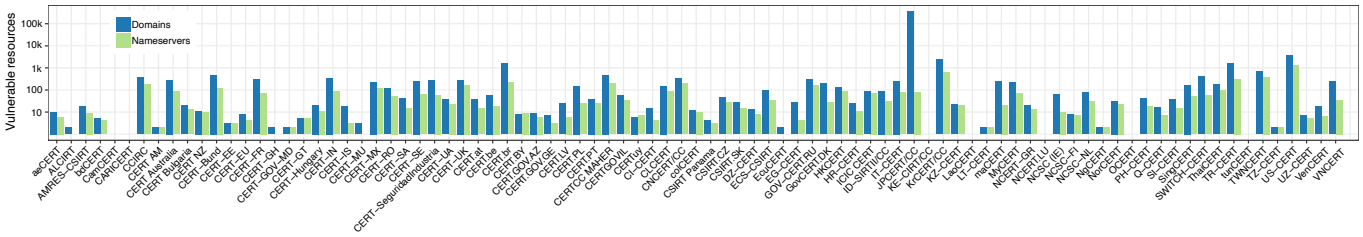

Figure 2. The number of vulnerable resources (domains and nameservers) per national CSIRT

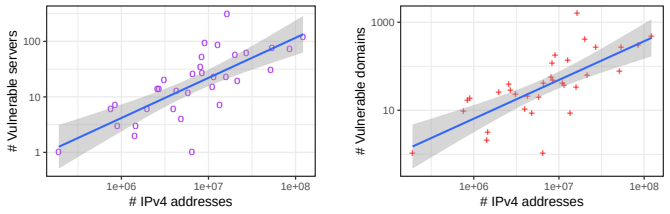

Figure 3. The distribution of vulnerable resources (domains and nameservers) per national CSIRTs with respect to their size (number of IPv4 addresses under their jurisdiction).

Table II
THE SUMMARY OF NOTIFICATION CAMPAIGNS

	Date	Notified	Unreachable	Replies
Phase 0	2017-05-01	1	-	1 manual
Phase 1	2017-09-06	44	2	7 automatic / 16 manual
	2017-09-28	35	2	6 automatic / 13 manual
	2017-10-19	27	2	4 automatic / 7 manual
Phase 2	2018-02-14	168	5	14 automatic / 40 manual
	2018-02-28	167	5	12 automatic / 24 manual
	2018-03-16	162	5	12 automatic / 24 manual
	2018-04-12	76	5	7 automatic / 7 manual

Remarkably, only three subdomains of vulnerable domains appeared in the Phishtank list. However, we found no evidence of their involvement in any abusive activities.

E. Descriptive Statistics of Vulnerable Resources

Given the high absolute number of vulnerable domains and nameservers, it is crucial to aggregate them so that we can further notify the affected entities more efficiently. After the introduction of GDPR [17], WHOIS databases provide very limited to no contact information about domain owners. Therefore, we explore the distribution of vulnerable domains and nameservers across autonomous systems (ASes) and Computer Security Incident Response Teams (CSIRTs).

1) Per-AS statistics: Table I shows that only 5.6K DNS nameservers are authoritative for the 382K vulnerable domains. On average, there are 121 domains per nameserver IP, but we identified three nameservers responsible for as many as 87% of all the vulnerable domain names. The AS distribution of vulnerable nameservers is diverse as they originate from 1,682 ASes. We note, however, that while four ASes host more than 100 vulnerable nameservers each, they do not translate into a large number of vulnerable domains. On the contrary, a single AS from Japan hosts nameservers responsible for 95.4% of vulnerable domains. As a result, while AS operators could potentially be our points of contact, it would require engaging with as many as 1.6K different entities.

2) Per-CSIRT statistics: Vulnerable nameservers are spread across 121 countries, with five of them (USA, South Korea, Taiwan, Turkey, and Iran) hosting almost half of all the vulnerable nameservers worldwide. Reporting vulnerabilities at the country level can be accomplished via CSIRTs. For each country hosting vulnerable nameservers and domains, we identified corresponding national CSIRTs (note that there can be multiple entities per country). We then computed the number of vulnerable domains and nameservers under the jurisdiction of each CSIRTs and plotted the result in Figure 2. Following the country-level pattern, the Japanese CSIRT is responsible for 2-orders of magnitude more vulnerable domains than any other CSIRTs. Aside from the JP-CERT, the distribution of vulnerable resources across national CSIRTs follows a log-normal distribution. Thus, while some CSIRTs have only one vulnerable resource, more than 50% of CSIRTs are responsible for hundreds of vulnerable domain names and nameservers.

To further investigate this log-normal distribution, Figure 3 plots: a) vulnerable authoritative nameserver IPs and b) vulnerable domain IPs (`A` records) to the IPv4 address space under the jurisdiction of each CSIRT. As expected, larger CSIRTs accumulate a higher number of vulnerable resources. Nevertheless, there are a few outliers. The Tunisian CSIRTs had only 1 vulnerable resource for a total of 6M IPv4 addresses. On the contrary, the Malaysian CSIRTs of similar size had more than 200 vulnerable resources.

V. NOTIFICATIONS

Non-secure dynamic updates pose a significant threat to domain owners. To motivate remediation action, we conducted a series of notification campaigns.

A. Notification Methodology

Choosing the right notification recipient ensures that the message will be understood and appropriate actions will be taken. We notified national CSIRTs—organizations responsible for reacting to cyber threats and ensuring the hygiene of networks under their jurisdiction. For each vulnerable resource, we identified the responsible entity and extracted the contact email address using CERT.at's database [21].

Table III
REMEDIATION EFFECTS BY THE END OF PHASE 1

	TF-CSIRTs (N = 44)		Other CSIRTs (N = 36)	
	Servers (Remediated/Vulnerable)	**Domain (Remediated/Vulnerable)**	**Servers (Remediated/Vulnerable)**	**Domains (Remediated/Vulnerable)**
N (%)	328 (20.4%) / 1280 (79.6%)	737 (14.24%) / 4439 (85.76%)	658 (19.17%) / 2775 (80.83%)	2173 (22.29%) / 7574 (77.71%)
max	62 / 235	108 / 1231	220 / 911	847 / 2384
median	3.0 / 14.5	4.0 / 29.5	2.5 / 13.0	4.5 / 34.0
mean (sd)	7.45 ± 11.71 / 29.09 ± 43.03	16.75 ± 27.41 / 100.89 ± 195.19	18.28 ± 39.10 / 77.08 ± 172.45	60.36 ± 151.84 / 210.39 ± 494.86

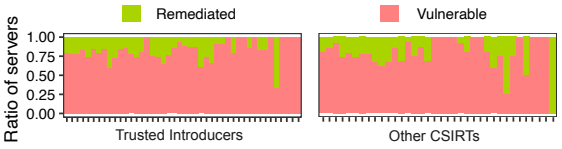

Figure 4. Remediation rates of the DNS nameservers during Phase 1 under two groups of CSIRTs - Trusted Introducers and Others

The subject line of our emails informs on the number of resources still vulnerable to zone poisoning. Specifically, it states "XX domain(s) still vulnerable to zone poisoning, YY nameservers fixed", where XX represents the number of vulnerable domains at the time of the notification and YY stands for the number of nameservers fixed since the global scan. It serves as a reminder nudge [46] to recall the recipients what is still to be fixed.

The body of the email contains four sections: i) a high-level description of the problem of non-secure dynamic updates, ii) the list of vulnerable nameservers and domain names, iii) names of organizations managing vulnerable resources, and iv) the list of necessary steps to fix the insecure configuration together with a pointer to a more extensive guide.

Table II summarizes the three phases of our notification campaign: *Phase 0* targeting the Japanese CSIRT, *Phase 1* targeting CSIRT members of the so-called "Trusted Introducer" community, and *Phase 2* targeting national and governmental CSIRTs. In total, we contacted 200 entities over six months.

B. Phase 0: JP-CERT

Given the high concentration of vulnerable resources in Japan, we contacted the JP-CERT before launching the large-scale notification campaign. While 29 nameservers were fixed and 92% of Japanese domains were not vulnerable anymore, the JP-CERT could not obtain more details from the affected network operators. To raise awareness in Japan and support the remediation action, JP-CERT wrote an article regarding zone poisoning and countermeasures.

After Phase 0, there remained more than 5K servers worldwide that accepted non-secure DNS updates from arbitrary clients, exposing more than 43K domains at risk of being exploited. We thus proceeded with the first phase of the notification campaign.

C. Phase 1: The Trusted Introducer Service

The Trusted Introducer Service (Task Force-CSIRT) [47] is a European CSIRT community formed in 2000. Its goal is to meet shared needs and construct a service architecture that provides critical assistance for all security and incident response teams. From September 7 to October 19, 2017, we sent emails to 44 teams that had vulnerable nameservers under their jurisdiction. While two CSIRTs were unreachable at indicated email addresses, around half of the remaining ones acknowledged the reception of our notifications either manually or via the creation of automatic tickets. We provided further information on the vulnerability to 5% of CSIRTs.

We then compare the remediation rates of the notified population against a random sample of 36 CSIRTs that were not part of the TF-CSIRT community at the time of the campaign. Table III presents the results at the end of Phase 1. Notified CSIRTs remediated 20.4% of corresponding nameservers and 14.24% of domains. We observe similar rates for the non-notified parties. This so-called "natural remediation" could be the result of i) transient misconfigurations that were detected independently, or ii) information received by the non-notified CSIRTs extraneous to our notification campaign.

Not all the CSIRTs contributed equally to the overall remediation. We tracked the population of vulnerable nameservers for three months and then computed the ratio of remediated machines to all those tested during the global scan. Figure 4 shows that while some CSIRTs fixed the considerable part of vulnerable nameservers, the average remediation rate per CSIRT was 26.2%. This highlights that CSIRTs are generally effective at remediating vulnerabilities but do not eliminate the problem completely.

We further compute the survival curves by using Kaplan-Meier estimates and interval censoring, defining the survival time as the time from the moment the notification was sent till the moment the resource was remediated. Figures 6 and 7 plot survival curves for vulnerable nameservers and domains, respectively. While their shape is similar, the Gehan-Breslow-Wilcoxon test indicates a significant difference between the curves. It suggests that our notifications were effective shortly after being sent and triggered better remediation rates compares to those CSIRTs that were not notified at all.

There exist different types of CSIRTs, e.g., governmental, national, military, research and education, non-commercial, and critical information infrastructure (CIIP). We wonder if the sector in which a CSIRT operates impacts the remediation rate. Figure 5 shows the survival curves for vulnerable nameservers with respect to the CSIRT type. We again refer to the Gehan-Breslow-Wilcoxon test to confirm that there is a statistically significant difference between the different types. Military

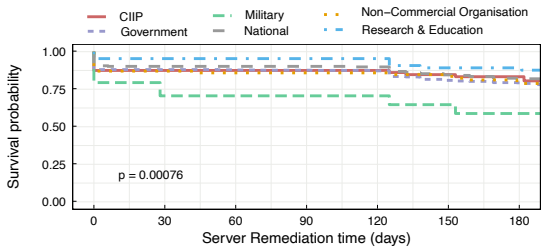

Figure 5. Server remediation rates of different CSIRT constituency types.

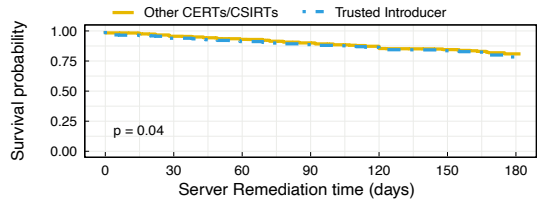

Figure 6. Survival rates of vulnerable nameservers during Phase 1.

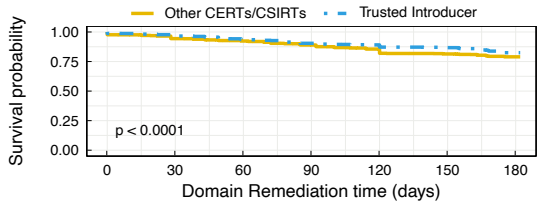

Figure 7. Survival rates of vulnerable domains during Phase 1.

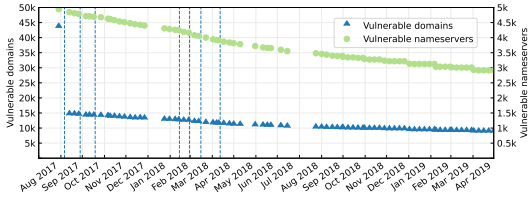

Figure 8. Number of vulnerable resources over time. The vertical dashed lines represent two notification campaigns - Phase 1 and Phase 2.

CSIRTs are particularly fast and efficient in remediation, contrary to research and educational entities that exhibit the slowest and worst patching rates.

D. Phase 2: National CSIRTs

We conducted the second phase of notifications three months after Phase 1. This time, we went beyond the Trusted Introducer Service and expanded our campaign to all the national CSIRTs with vulnerable resources under their jurisdiction. We also modified the notification email and included a link to our website containing all the aggregated statistics about the number of vulnerable resources globally. This allowed all visitors to track their mitigation progress over time, but also to compare themselves against the rest of the CSIRTs. We hypothesized that this social norms nudge [46] would encourage CSIRTs to fix the vulnerability.

In total, we notified 168 national CSIRTs. Figure 8 plots the number of vulnerable resources observed during Phase 1 and Phase 2 notification campaigns. The vertical dashed lines refer to the dates when the emails were sent—three days in 2017 for Phase 1 and four days in 2018 for Phase 2. This second notification campaign accelerated the remediation rates, although not all the CSIRTs were equally efficient. Around 2% of national CSIRTs reached back to inform us that acting upon this vulnerability was not within their mandate.

Overall, the three notification campaigns led to fixing 97.96% of vulnerable domains and 53.59% of vulnerable nameservers. These remediation rates are higher than the ones reported by previous studies and signal the need to involve CSIRTs in the remediation of vulnerabilities.

VI. Long-Term Impact

We now assess the lasting impact of our three notification campaigns on securing domains and nameservers from zone poisoning. Four years later, we perform active measurements to gauge the increase in newly vulnerable domain names and the persistence of previously identified vulnerable resources.

Our input dataset was collected in June 2022 and comprised the following sources: i) Tranco domain popularity list [38], ii) Certificate Transparency logs from Calidog [6], iii) passive DNS query traces from SIE Europe [42], iv) 1150 legacy gTLD and new gTLD zone files provided by ICANN's CZDS [18], v) AXFR transfers for `.se`, `.nu`, `.ch`, and `.li` zones. During this campaign, we aggregated all the domain lists and queried `SOA` records to only keep active domain names for further analysis. We then requested nameserver resource records (`NS`) and their corresponding IP addresses (`A` and `AAAA` records). Moreover, we extended our measurements to IPv6 address space.

The data sets for both the 2017 and 2022 scans are comparable, including the vast majority of domain names under gTLDs (`.com`, `.org`, `.net`, etc.), along with valuable passive DNS datasets that aid in identifying ccTLD domains. The distinction is that in the 2022 scans, we first identify registered domains through active measurements and then conduct our experiments. However, in the 2017 scans, the data includes both registered and expired domain names from our input list, and we subsequently identify vulnerable registered domains.

In total, we sent 1.4 billion update requests for 260M domains and 971K nameserver addresses (both IPv4 and IPv6). The number of successful updates is low. In IPv4 address space, 6,478 successful DNS updates impacted 5,495 domains and 2,072 nameservers. Moreover, a considerable fraction of those (21.4% of domains and 23.6% of nameservers) were not fixed since the 2017 global scan. While very few new resources became vulnerable after our notification campaigns, others were insecure for years. Much fewer domains (168) were vulnerable when sending updates to corresponding nameservers over IPv6—73 nameservers being authoritative for 68 domains accepted our updates.

To gain deeper insights into the popularity of vulnerable domain names, we compiled the Alexa top website lists for

the year 2022. The results showed a significant reduction, with only 516 domain names identified as vulnerable, compared to the staggering 5,964 vulnerable domains detected back in 2017—a difference of two orders of magnitude. Moreover, among the 200,000 most popular domain names, we found a mere 172 vulnerable domains. Our evaluation of the notification campaigns presents a positive overall picture, underscoring the effectiveness of our efforts in securing a substantial portion of domains and nameservers from zone poisoning attacks over the long term.

Finally, in Section IV-D, we endeavored to identify indications of zone poisoning attacks in passive DNS and blocklists during 2017. In this section, we propose an alternative methodology and deploy honeypots with a vulnerable configuration to attract potential attackers and detect signs of the attacks in the present day. Specifically, we installed BIND9 as an authoritative nameserver accepting non-secure updates. It hosted 105 domain names, 50 being drop-catch (recently expired) domains appearing in the Alexa top list. We generated full-fledged websites for each hosted domain and obtained TLS certificates. We performed a zone poisoning attack ourselves to ensure it was feasible and managed to take over the domains by updating their `A` records. Nevertheless, despite attracting numerous port scanners and attackers, the honeypot received no unsolicited DNS `UPDATE` requests within seven months.

VII. Ethical Considerations

Active Internet measurements have become an established practice in computer networking research. The community has developed a set of best practices to guide researchers in conducting measurements ethically [12], [14]. In our research, we rigorously adhered to the guidelines set forth in the Menlo report [12], and the ethical considerations outlined by Korczyński et al. [23] in their preliminary research on zone poisoning. These guidelines and considerations provided us with a solid framework to ensure the ethical conduct of our study.

We created dedicated web pages on each inserted `researchstudyzp` subdomain, providing clear details about the purpose of our scan and offering contact information. Despite the extremely low probability of a collision with an existing subdomain, we thoroughly assessed this scenario. It should be noted that the newly added record would not overwrite the existing one. All the records were removed after the study. Throughout the process, we received emails from several network operators seeking further details, requesting information about our methodology, and confirming the fixes they had applied.

Furthermore, we prioritize the principle of beneficence, aiming to benefit the community by identifying vulnerable systems with minimal intervention and taking steps to notify and assist their owners. Our commitment to responsible disclosure involves sharing our findings exclusively with the relevant parties: the authors of the original RFC, CSIRTs, and DNS software vendors, rather than publicly exposing identifiable information about vulnerable resources.

To ensure the highest ethical standards, our research proposal, and accompanying materials, including our approach to conducting active internet scans and notifying CSIRTs, underwent a rigorous review by the human research ethics committee at our institution. The committee carefully evaluated potential risks and concerns, ensuring that appropriate measures were in place to protect the privacy and confidentiality of the domain owners involved.

VIII. Related Work

Researchers rely on large-scale notifications to inform parties about vulnerabilities (e.g., [28], [35], [48], [49]). However, choosing the appropriate channel, sender, recipient, and framing is challenging.

Recipient selection is crucial for effective notification. In a study targeting websites with WordPress vulnerabilities, reaching owners directly and indirectly resulted in an overall remediation rate of no more than 26.5% [45]. GDPR regulations later made it difficult to retrieve contact emails from domain WHOIS, prompting researchers to find alternative means of reaching domain owners. `SOA` records were found to be a useful source for administrator emails [44], and CERTs were utilized for dissemination [26]. Comparing remediation rates between CERT notifications and direct WHOIS notifications showed better performance in the latter group [27].

Previous studies focused on non-secure dynamic updates and identified vulnerable domains, but did not perform notifications [23]. In subsequent work, contacting owners via `SOA`, generic, and WHOIS email addresses resulted in low remediation rates [7]. Our research builds upon these studies by introducing an extensive adversary model and conducting Internet-wide measurements, revealing significantly more vulnerable domains [23]. Employing an indirect approach to notifications, we achieved higher remediation rates, with 54% of nameservers and 98% of domains being remediated [7].

IX. Conclusions

In this paper, we analyzed non-secure DNS updates—a standard that lets anyone update the content of DNS zones. We defined an extensive attack taxonomy that shows how a single DNS update packet can enable an attacker to make a domain unavailable, take it over, or compromise the domain control validation.

We performed a large-scale analysis of more than 354M domains and 3.8M corresponding nameservers. Less than 1% of vulnerable domains and nameservers, including those of financial, governmental, and healthcare institutions, were identified. We notified national CSIRTs and achieved remediation rates of approximately 98% for domains and 54% for nameservers. Our repeated scans in 2022 confirmed the long-term impact of our notifications, with a low population of vulnerable resources.

Efforts to fix individual systems are labor-intensive and do not provide a guarantee against the recurrence of insecure configurations in the future. Hence, we engaged in discussions with Paul Vixie, the original author of RFC 2136, as well as

DNS software vendors, to share the findings of this paper. At the time of writing, two CVEs have been reserved for the DNS vendor software, specifically targeting Knot DNS and Simple DNS Plus.

Acknowledgment

The authors express their gratitude to Paul Vixie, CSIRTs, and DNS software vendors for their valuable comments and active participation in the mitigation of vulnerable systems. The authors thank the contributors of passive DNS data to Farsight Security and the European Data Sharing Collective (SIE Europe). This work has been partially supported by Carnot LSI and Grenoble Alpes Cybersecurity Institute (under the contract ANR-15-IDEX-02), the French Ministry of Research projects PERSYVAL-Lab under contract ANR-11-LABX-0025-01, and DiNS under contract ANR-19-CE25-0009-01.

References

[1] Donald E. Eastlake 3rd. Secure Domain Name System Dynamic Update. RFC 2137, 1997.

[2] Donald E. Eastlake 3rd, Ólafur Guðmundsson, Paul A. Vixie, and Brian Wellington. Secret Key Transaction Authentication for DNS (TSIG). RFC 2845, 2000.

[3] AerServ. List Of IAB Categories, 2023. https://support.aerserv.com/hc/en-us/articles/207148516-List-of-IAB-Categories.

[4] Amazon. Alexa: Actionable Analytics for the Web, March 2022. https://www.alexa.com.

[5] APWG. Unifying The Global Response To Cybersrime, March 2022. https://apwg.org.

[6] Cali Dog Security. Certstream, March 2022. https://calidog.io.

[7] Orcun Cetin, Carlos Gañán, Maciej Korczyński, and Michel van Eeten. Make Notifications Great Again: Learning How to Notify in the Age of Large-scale Vulnerability Scanning. In *WEIS*, 2017.

[8] Yizheng Chen, Manos Antonakakis, Roberto Perdisci, Yacin Nadji, David Dagon, and Wenke Lee. DNS Noise: Measuring the Pervasiveness of Disposable Domains in Modern DNS Traffic. In *IEEE DSN*, 2014.

[9] Stuart Cheshire and Marc Krochmal. DNS-Based Service Discovery. RFC 6763, 2013.

[10] Cisco Talos Intelligence Group . PhishTank, March 2022. https://phishtank.org.

[11] CZ.NIC. Knot DNS 3.1.9 documentation, July 2023. https://www.knot-dns.cz/docs/3.1/singlehtml/.

[12] David Dittrich and Erin Kenneally. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. Technical report, U.S. Department of Homeland Security, 08 2012.

[13] Ralph Droms. Dynamic Host Configuration Protocol. RFC 2131, 1997.

[14] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *USENIX Security*, August 2013.

[15] Farsight Security. DNS Database (DNS-DB), March 2016. https://www.dnsdb.info.

[16] The Swedish Internet Foundation. Access to zonefiles for .se and .nu, March 2022. https://www.iis.se/english/domains/tech/zonefiles/.

[17] ICANN. Temporary Specification for gTLD Registration Data, 2018.

[18] ICANN. Centralized Zone Data Service, March 2022. https://czds.icann.org.

[19] Internet Systems Consortium. BIND 9 Administrator Reference Manual, July 2022. https://bind9.readthedocs.io/en/v9_18_4/index.html.

[20] JH Software. Simple DNS Plus, July 2022. https://simpledns.plus/help/definition-dynamic-dns-update.

[21] L. Aaron Kaplan. CERT.at geolocate the national CERT abuse service, 2023. https://contacts.cert.at/cgi-bin/abuse-nationalcert.pl.

[22] Scott Kitterman. Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1. RFC 7208, 2014.

[23] Maciej Korczyński, Michał Król, and Michel van Eeten. Zone Poisoning: The How and Where of Non-Secure DNS Dynamic Updates. In *IMC*, 2016.

[24] Murray Kucherawy, Dave Crocker, and Tony Hansen. DomainKeys Identified Mail (DKIM) Signatures. RFC 6376, 2011.

[25] Murray Kucherawy and Elizabeth Zwicky. Domain-based Message Authentication, Reporting, and Conformance (DMARC). RFC 7489, 2015.

[26] Marc Kührer, Thomas Hupperich, Christian Rossow, and Thorsten Holz. Exit from Hell? Reducing the Impact of Amplification DDoS Attacks. In *USENIX Security*, 2014.

[27] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami, Michael Bailey, Damon McCoy, Stefan Savage, and Vern Paxson. You've Got Vulnerability: Exploring Effective Vulnerability Notifications. In *USENIX Security*, 2016.

[28] 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 *WWW*, 2016.

[29] Linux man page. nsupdate(8) - Linux man page, 2023. https://linux.die.net/man/8/nsupdate.

[30] Daiping Liu, Zhou Li, Kun Du, Haining Wang, Baojun Liu, and Haixin Duan. Don't Let One Rotten Apple Spoil the Whole Barrel: Towards Automated Detection of Shadowed Domains. In *CCS*, 2017.

[31] Qasim Lone, Matthew J. Luckie, Maciej Korczyński, and Michel van Eeten. Using Loops Observed in Traceroute to Infer the Ability to Spoof. In *PAM*, pages 229–241, 2017.

[32] Paul Mockapetris. Domain names - concepts and facilities. RFC 1034, 1987.

[33] Paul Mockapetris. Domain names - implementation and specification. RFC 1035, 1987.

[34] Mozilla Foundation. Publix Suffix List, March 2022. https://publicsuffix.org.

[35] Antonio Nappa, M. Zubair Rafique, and Juan Caballero. Driving in the Cloud: An Analysis of Drive-by Download Operations and Abuse Reporting. In *DIMVA*, 2013.

[36] Yevheniya Nosyk, Qasim Lone, Yury Zhauniarovich, Carlos Hernandez Gañán, Emile Aben, Giovane C. M. Moura, Samaneh Tajalizadehkhoob, Andrzej Duda, and Maciej Korczyński. Intercept and Inject: DNS Response Manipulation in the Wild. In *PAM*, 2023.

[37] Paul Pearce, Ben Jones, Frank Li, Roya Ensafi, Nick Feamster, Nick Weaver, and Vern Paxson. Global Measurement of DNS Manipulation. In *USENIX Security*, 2017.

[38] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *NDSS*, 2019.

[39] PowerDNS. Dynamic DNS Update (RFC2136), July 2022. https://doc.powerdns.com/authoritative/dnsupdate.html.

[40] Rapid7 Labs. Internet-Wide Scan Data Repository: DNS Records (ANY), March 2016. https://scans.io/study/sonar.fdns.

[41] SIDN. Stichting Internet Domeinregistratie Netherland (SIDN) Labs, March 2022. https://www.sidn.nl.

[42] SIE Europe. Passive DNS Data Sharing, March 2022. https://www.sie-europe.net.

[43] Sooel Son and Vitaly Shmatikov. The Hitchhiker's Guide to DNS Cache Poisoning. In *Security and Privacy in Communication Networks*, 2010.

[44] Wissem Soussi, Maciej Korczyński, Sourena Maroofi, and Andrzej Duda. Feasibility of Large-Scale Vulnerability Notifications after GDPR. In *IEEE EuroS&PW Workshops*, 2020.

[45] 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*, 2016.

[46] C. Sunstein. Nudging: A Very Short Guide. *Journal of Consumer Policy*, Volume 37, Issue 4, pp 583-588, 11 2014.

[47] TF-CSIRT Trusted Introducer. Services for Security and Incident Response Teams, 2023. https://www.trusted-introducer.org.

[48] Marie Vasek and Tyler Moore. Do Malware Reports Expedite Cleanup? An Experimental Study. In *CSET*, 2012.

[49] Marie Vasek, Matthew Weeden, and Tyler Moore. Measuring the Impact of Sharing Abuse Data with Web Hosting Providers. In *WISCS*, 2016.

[50] Verisign. Top-Level Domain Zone File Information, March 2022. https://www.verisign.com/en_US/channel-resources/domain-registry-products/zone-file/index.xhtml.

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

[52] Webshrinker. Domain & Threat Data, 2023. https://www.webshrinker.com.

[53] Brian Wellington. Secure Domain Name System (DNS) Dynamic Update. RFC 3007, 2000.