Measuring Exposure in DDoS Protection Services

Mattijs Jonker (✉), Anna Sperotto
University of Twente, Enschede, The Netherlands
{m.jonker, a.sperotto}@utwente.nl

***Abstract*—Denial-of-Service attacks have rapidly gained in popularity over the last decade. The increase in frequency, size, and complexity of attacks has made DDoS Protection Services (DPS) an attractive mitigation solution to which the protection of services can be outsourced. Despite a thriving market and increasing adoption of protection services, a DPS can often be bypassed, and direct attacks can be launched against the origin of a target. Many protection services leverage the Domain Name System (DNS) to protect, e.g., Web sites. When the DNS is misconfigured, the origin IP address of a target can leak to attackers, which defeats the purpose of outsourcing protection. We perform a large-scale analysis of this phenomenon by using three large data sets that cover a 16-month period: a data set of active DNS measurements; a DNS-based data set that focuses on DPS adoption; and a data set of DoS attacks inferred from backscatter traffic to a sizable darknet. We analyze nearly 11k Web sites on Alexa's top 1M that outsource protection, for eight leading DPS providers. Our results show that 40% of these Web sites expose the origin in the DNS. Moreover, we show that the origin of 19% of these Web sites is targeted after outsourcing protection.**

I. Introduction

Denial-of-Service (DoS) attacks aim to disrupt legitimate services, thereby causing harm to the service operator and legitimate users. In the past years we have witnessed an increase in occurrence and strength of DoS attacks, with recent reports of attacks reaching 1Tbps [1]. Events such as the attack against the Domain Name System root [2], or against the service and DNS provider Dyn [3], have shown that DoS attacks can provoke extensive damage by attacking core Internet infrastructure (e.g., the DNS).

The rise of DoS attacks has created a market for DDoS Protection Services (DPSs). The protection of a specific application (e.g., a Web site) or even an entire network can be outsourced. Protection can take place on customer premises, by means of in-line appliances [4]. Protection can also take place in the cloud, where malicious traffic is filtered and absorbed. Hybrid solutions also exist, where customer premises equipment is combined with a cloud-based component. Large attacks, i.e., those high in network traffic volume, are best mitigated in the cloud. The key mechanism to outsource protection to a DPS is *traffic diversion*, i.e., routing network traffic to the security infrastructure of the DPS. One way to divert traffic to services that are reached on the basis of a domain name is to leverage the DNS, similarly to how load balancing is achieved in content delivery networks [5], [6].

DNS-based network traffic diversion requires the origin of a service (e.g., a Web server's actual IP address) to only be known to, or accessible by, the DPS provider. In case the origin is known to potential attackers, the DPS can be bypassed to launch direct attacks [7]. There are various means by which potential attackers can determine the origin of a service. For example, various DNS Resource Records (RRs) may leak the origin's IP address. In 2013, Nixon and Camejo drew attention to this phenomenon [8]. In 2015, Vissers et al. performed an assessment at scale [9].

The goal of this paper is to quantify the extent to which DPS-protected Web sites are susceptible to direct DoS attacks on their infrastructure. We make three contributions to earlier work. First, we leverage a large data set of active DNS measurements to analyze origin exposure in the DNS at a larger-than-ever-before scale. Our analysis targets Web sites on Alexa's top 1M and covers eight leading DPS providers for which, in recent work, we have shown an increasing trend in adoption [10]. Second, our analysis encompasses a comprehensive set of exposure vectors among which two novel, not previously investigated vectors. Third and final, we correlate exposed origin IP addresses with a large data set of DoS attack events. This allows us to infer if attacks on the origin of Web servers take place after Web sites start outsourcing protection. We use various large data sets for our analyses, all of which cover the same 16-month period (Jan 2016 – May 2017).

Our analysis shows that *40.5%* of the Web sites that outsource protection for at least three months expose their origin in the DNS. We also show that DoS attacks target the origin IP addresses of *19%* of all analyzed Web sites after they start outsourcing protection.

Section II contains background information on DoS attacks and protection services. In Section III we describe various vectors through which Web sites can reveal the origin in the DNS. Section IV describes our methodology and Section V details the data sets that we used. Our results are presented in Section VI. Section VII contains related work. Finally, our conclusions are in Section VIII.

II. Background

A. Denial-of-Service Attacks

The goal of Denial-of-Service (DoS) attacks is to deny access to a networked service. DoS attacks oftentimes achieve this by means of resource exhaustion, which can take place at the network level (e.g., link saturation) or at the server level (e.g., overload a daemon with requests). Attack traffic is sent towards a victim *directly* or by means of *reflection*. *Direct* traffic is sent directly from a source or set of sources under control by the attacker to the target, e.g., from the attacker's

978-3-901882-98-2

computer, a compromised server, or a set of compromised machines (e.g., a botnet). To hinder forensics, and to complicate the mitigation of attacks, attackers typically spoof the source IP address in direct traffic to a random, fake address. In *reflection* attacks, a third-party service (i.e., the reflector) is abused to reflect traffic towards the target. Reflection also involves spoofing, but not the random kind. The source IP address of a request is set to the target's IP address by the attacker. The reflector, which has no way of distinguishing legitimate from malicious requests, sends its responses to the target. Many protocols that allow for reflection attacks also send a response that is much larger than the request, which means that traffic is amplified [11]. Amplification does not just affect older protocols such as NTP and IGMP [12], [13], but also newer protocols such as DNSSEC [14]. DoS attacks are referred to as Distributed Denial-of-Service (DDoS) attacks in case traffic is sent to the target from a distributed set of sources (e.g., a botnet or a set of reflectors).

There are two classes of DoS attacks: *volumetric* and *semantic* attacks. Volumetric attacks exhaust resources by means of sending massive traffic volumes. These attacks are typically service-agonistic. In contrast, semantic attacks are service-specific and try to exploit flaws (e.g., protocol vulnerabilities) to deny access to a networked service. Unlike volumetric attacks, semantic attacks have negligible bandwidth effects.

B. DDoS Protection Services

DDoS Protection Services (DPSs) offer victims of attacks a means to outsource protection. A DPS can offer various types of mitigation solutions to deal with volumetric attacks, semantic attacks, or both. Solutions can rely on in-line appliances on customer premises, require network traffic to be diverted to the cloud (i.e., the security infrastructure of the DPS), or be a hybrid of both. Large volumetric attacks are likely to saturate the customer's network link and thus require traffic diversion, whereas semantic attacks can be mitigated in-line given the minimum bandwidth effects [15]. In this paper we focus on DPS use by Web sites by means of network traffic diversion (i.e., all but strictly in-line mitigation). Network traffic diversion is commonly achieved through the Domain Name System (DNS), or by using the Border Gateway Protocol (BGP).

With **DNS-based traffic diversion**, the DNS is leveraged for traffic diversion similar to how content delivery networks implement load-balancing [5]. A DNS-based setup is typically combined with a reverse proxy that is placed between the *origin* of a service and potentially malicious requests to that service. The proxy forwards only benign requests to the origin – which sits outside of the DPS infrastructure – and serves responses from within the DPS infrastructure. This is a form of security by obscurity: all clients of a service (both malicious and legitimate) are expected to use the DNS to resolve a domain name to an IP address (i.e., the proxy's) in the DPS infrastructure. The origin IP address needs only to be accessed by the reverse proxy and is to remain obscured from attackers. If this obscurity fails, the DNS may be bypassed, and DoS attacks can be launched on the origin directly [8], [7].

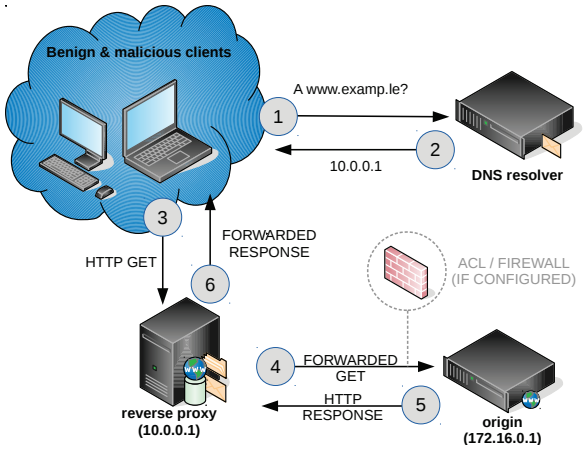

Fig. 1: Schematic of DNS-based network traffic diversion

Figure 1 shows a DNS-based diversion setup. The DNS of the Web site `www.examp.le` is setup to resolve to `10.0.0.1`, which is the IP address of the reverse proxy that is located within the DPS infrastructure. Any client looking to make a Web request (i.e., `GET`) should speak to `10.0.0.1`. Only `10.0.0.1` should speak to the origin Web server at `172.16.0.1`, which can be enforced with a properly configured firewall.

There are various ways in which the DNS can be leveraged to setup network traffic diversion. First, the IP address Resource Record (RR) of `www.examp.le` (i.e., its `A` RR) can be configured to `10.0.0.1`.

```
www.examp.le   IN  A   10.0.0.1
www.examp.le   IN  NS  ns.registr.ar
```

Second, `www.examp.le` can be made into an alias for another domain name by setting a `CNAME` RR. This alias is "expanded" to an `A` record that is set to `10.0.0.1`. The subtle difference with the first case is that it is the aliased domain that has the ability to set the `A` record, and not `www.examp.le`. Typically, the aliased domain (`foob.ar` in the example shown below) belongs to the DPS.

```
www.examp.le   IN  CNAME   foob.ar
foob.ar        IN  A       10.0.0.1
foob.ar        IN  NS      ns.foob.ar
```

The third case involves changing the authoritative name server (i.e., the `NS` RR) of `examp.le`. If the DPS controls the `NS` (`ns.foob.ar` in the example below), it answers all queries for RRs of `www.examp.le`, including the `A` record that points to `10.0.0.1`.

```
www.examp.le   IN  A   10.0.0.1
www.examp.le   IN  NS  ns.foob.ar
```

When DNS-based diversion is used, it is recommended to drop requests to the origin from any source but the DPS proxy, or a set of proxies [16]. Properly configuring a firewall up can be neglected (see Figure 1), leaving the origin Web server vulnerable to attacks. Moreover, in some cases, setting up a firewall is a complicated or even an infeasible endeavor if a large number of reverse proxies are used by a DPS [17].

With **BGP-based traffic diversion**, the DPS announces a customer-used prefix, e.g., a /24, to divert all customer-destined traffic. Any traffic sent to an IP address in the announced prefix passes through the DPS infrastructure, where malicious traffic can be filtered. Clean traffic is returned to the customer's network using, e.g., a Generic Routing Encapsulation (GRE) tunnel. In a BGP setup, the *origin* need not be obscured, as any traffic sent to the origin's IP address directly will still be routed towards the DPS, which is why we do not investigate BGP-based DPS use in this paper.

Diversion can be *always-on* or done in an *on-demand* manner. With the prior, traffic is always diverted to the DPS, even if no attacks are active. With *on-demand* protection, a DNS change or BGP prefix announcement is made in response to an attack, and negated when mitigation has completed.

The type of customer and types of attacks weigh in on the potential of either diversion setup. A customer that wishes to protect an entire network block (e.g., multiple Web servers with multiple IP addresses) may want to use BGP. In contrast, the owner of a single machine (or even a single service instance on a machine) can use a DNS-based setup. A DNS-based setup is typically easier to configure (if done right) and requires fewer resources (e.g., you do not need to configure BGP). The downsides of a DNS-based setup are that it only works for proxiable applications (e.g., Web sites) and the fact that it can be bypassed to launch direct attacks. In this paper we focus on exactly those drawbacks: we reveal the origin in DNS-based setups on the basis of large-scale DNS measurements, and then correlate the origin IP address with a large data set of DoS attack events.

III. Exposure Vectors

In Section II-B we explained that if security by obscurity in DNS-based traffic diversion is broken, direct DoS attacks may be launched on the origin. In this section we explain various vectors through which the origin can be exposed. We describe two novel vectors (Sections III-D and III-F) in addition to various vectors that were previously identified [9], [7], [8].

A. Third-Level Domains (3LD)

The label www is typically used to give a canonical name to Web sites. Consider www.example.com, which is a third-level domain (3LD) in the zone of example.com. A zone can contain a number of labels, at various levels. Domain name administrators can configure labels such as ftp and mail to give canonical names to, respectively, *FTP* and *SMTP* servers. If other services run on the same IP address as the origin Web server, then the IP address of the origin may be exposed through the Resource Record (RR) on non-www labels.

This may seem like a trivial error to make, but given that not all protocols are interoperable with the reverse proxy mechanism on which DPS providers rely, it is not uncommon to have labels pointing directly to customer infrastructure.

Techniques to reveal other labels include: simply brute-forcing them, using zone enumeration [18], or by monitoring queries for them (e.g., using passive DNS [19]). In addition, certain DPS providers are associated with predictable labels to the origin. A first example are labels in DPS documentation that people copy verbatim while configuring the DNS for their domain.[1] Another example are labels that are (automatically) added to for non-proxiable protocols (see Section III-B).

B. Mail Exchanger (MX)

The MX record of a domain name specifies the location of the domain's mail server. The name in the MX RR typically resolves to an IP address that is running a *Simple Mail Transfer Protocol* (SMTP) server. Mail can be dealt with by a mail provider such as *Google Mail*, but domains can also host their own mail server. In the latter case, as DPS providers do not proxy mail ports, the MX RR will resolve to a customer IP address. If the mail server runs on the same IP address as the origin Web server, the MX RR exposes the origin.

If a domain hosts its own mail server, the MX RR can specify a name within the domain's own DNS zone. In other words, the MX of examp.le could point to mail.examp.le. This necessitates a 3LD label, as explained in Section III-A.[2] An example is shown below. The www label resolves to the DPS proxy, whereas the mail label exposes the origin.

```
www.examp.le  IN  A   10.0.0.1
examp.le      IN  MX  mail.examp.le
mail.examp.le IN  A   172.16.0.1
```

C. Sender Policy Framework (SPF)

The Sender Policy Framework (SPF) [20] allows domain name administrators to combat forged e-mails that appear to originate from a domain but are in fact sent by a rogue or compromised mail server with no relation to the domain at hand. This is done by publishing SPF information in the zone of a domain by means of a TXT record with an SPF value indicator (i.e., *v=spf1 ...*). This record can specify, among others, the IP addresses from which e-mail can legitimately originate. If the domain's e-mail server runs on the same IP address as the Web server, the origin is leaked. A simple example is below, where the specified IP address exposes the origin (recall, 10.0.0.1 is the reverse proxy).

```
www.examp.le IN  A   10.0.0.1
examp.le     IN  TXT "v=spf1 ip4:172.16.0.1 -all"
```

D. Name Server (NS)

The NS record specifies the location of the name server that is authoritative for a domain. This name server could be in-bailiwick, meaning that the NS record for examp.le could point to, e.g., ns.examp.le. If the name server runs on the same IP address as the Web server, the origin is leaked. A self-explanatory example is shown next.

```
www.examp.le  IN  A   10.0.0.1
examp.le      IN  NS  ns.examp.le
ns.examp.le   IN  A   172.16.0.1
```

[1] An example of this origin-www from Akamai's documentation.

[2] CloudFlare formerly automatically configured the direct-connect label to avoid potential conflicts with MX records. Nowadays, it will be set to dc-<rand>.example.com, which is harder to guess by brute-forcing 3LDs, but will still leak from MX records.

E. Conflicting Records

Web site administrators may, while configuring DNS for diversion, neglect to remove IP address records that point to the origin. This could lead, e.g., to two `A` RRs on the `www` label, one pointing to the origin, and the other to the proxy. A typical example is where the root of a domain (i.e., its apex, or `@`) still points to the origin, while the `www` label is properly set to the DPS proxy. A self-explanatory example is shown below. We refer to these cases as "conflicting records".

```
www.examp.le  IN  A   10.0.0.1
www.examp.le  IN  A   172.16.0.1
```

F. IPv6 Address (AAAA)

Web servers may be dual-stacked, which means they can be reached over IPv6 as well as over IPv4. Not many DPS providers support IPv6, which means the `AAAA` record of a `www` label could expose the origin, even if IPv4 traffic is properly diverted to the DPS. In these cases the origin might be attacked using its IPv6 address.

G. IP Address History

The exposure vectors explained thus far relate to the way the zone of a Web site is setup at the moment a prospective attacker inspects the current state of the DNS. While the current state may no longer expose a Web site's origin, historic DNS data still can. For this reason, administrators are recommended to use a "clean" IP address for the origin, i.e., one that is not publicly known once they have configured DNS-based traffic diversion [21]. Using the old IP address breaks obscurity more easily.

H. Is an IP address a sufficient indication of exposure?

A leaked IP address is not per se an indication of exposure. It might be that an address no longer corresponds to the origin, or requests from anything but the reverse proxy are filtered. For this reason an extra step needs to be taken to make sure that the origin has been found. We address this in our methodology.

IV. Methodology

The methodology by which we study the extent to which Web sites expose their origin IP address through the DNS contains various steps. Figure 2 shows the overall flow of information in our methodology. We start with a selection of DPS-using Web sites. For these Web sites we find potential origin exposure in the DNS using a longitudinal data set of active DNS measurements. We then take the potential origin IP addresses and scrape them for Web content, both by using the DNS (i.e., connecting to the DPS proxy), and by bypassing the DNS (i.e., connecting directly). We then apply a custom DOM-tree comparison method to see if retrieved Web content is similar, i.e., the potential origin in fact corresponds to the origin. We further detail these steps in the following sections.

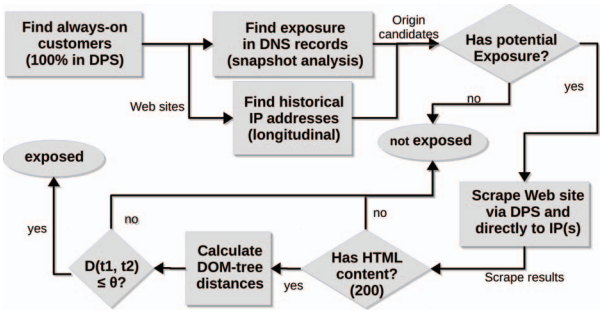

Fig. 2: Steps taken to analyze if Web sites expose their origin IP address through the DNS

A. Long-Term, Always-On Protected Web Site Selection

Our analysis starts with a selection of long-term, always-on Web site customers for a selection of DPS providers. We look at always-on customers only because on-demand customers reveal the origin by design when traffic diversion is not active. Our selection targets Web sites that have been a customer for at least three months on the day that we perform our analysis. We take this selection as representative for the customers of a DPS that have a stake in hiding their origin. We select these Web sites using the methodology previously published in [10]

B. Deriving Potential Origin IP Addresses

Given a set of always-on customers we pull relevant DNS resource records from a data set of long-term, active DNS measurements. If the DNS of a Web site is configured as such that the origin is leaked, the data set will reflect it.

Given `examp.le`, we pull IP addresses (i.e., `A` & `AAAA`) for the labels of interest: `www`, `@`, and for all the labels on the list of commonly-used labels mentioned in Section V-A. We also pull IP addresses for the names in `examp.le`'s `MX` and `NS` records. Moreover, we extract full IP addresses from `TXT` records that contain SPF information.[3]

We also pull historical IP addresses from the data set, on days that precedes these day on which `www.examp.le` became DPS customer. We filter all IP addresses that are invalid (e.g., `400.3.2.1`), within private address space (e.g., `192.168.0.0/16`), or those that are routed to an autonomous system (AS) number of a DPS provider.

C. Scraping Potential Origin Addresses for Content

Our set of always-on Web sites and potential origin addresses is fed to a scraper, which sends "regular" requests to fetch content through the DPS proxy. Moreover, our scraper bypasses the DNS and sends "direct" requests to IP addresses that potentially correspond to the origin. Our scraper is built in Python, uses multi-threading, and has a requests pacing and backoff mechanism to avoid stressing hosts with repeated connection attempts or Web requests. All results (e.g., HTTP

[3]We only consider full addresses in SPF information and not network blocks, e.g., a `/24`. Furthermore, SPF includes are not followed, because these typically lead to third-party IP addresses.

status codes, connection errors, and content) are stored. Any content accompanied by an `OK` status code is fed to our origin verification system once all scraping has completed. For other status codes we infer that the potential IP address does not correspond to the origin.

D. Origin Verification

An `OK` status code on Web content does not imply that we found the origin. To verify if the origin was found, we make, for any given Web site, pairwise comparisons between the content that was returned by the DPS proxy and that of each of the requests that bypassed the proxy. As Web site content can vary with every page load (e.g., ads or dynamic content such as page generation timestamps) we cannot simply do a one-to-one comparison between the contents of two requests. Instead, we account for variable change by relying on a DOM-tree comparison, the motivation for which is that variable change in the content on a Web site modifies some, but not all of its DOM-tree structure.

We base our comparison on the the tree-edit distance algorithm by Zhang and Shasha [22], which counts the number of edit operations (inserts, deletes, and substitutions) to get from one tree to another. Rosiello et al. [23] used this algorithm to compare phishing Web sites to their original, and Vissers et al. [9] use it in a work similar to ours.

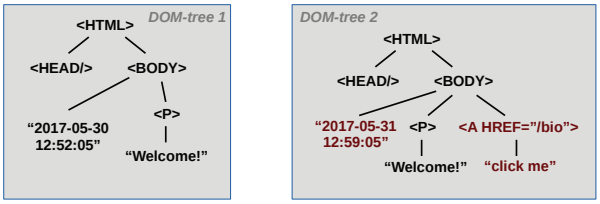

Fig. 3: Two simple DOM trees to compare for distance

As an example of how the distance algorithm works in its pure form, consider the two trees in Figure 3. The labels of non-leaf nodes correspond to HTML or XML elements and the leaf labels correspond either to strings or empty elements. The tree's parent-child relations correspond to nesting. Text leafs are formed by text on Web pages (e.g., in a link) or by comments in the HTML source (i.e., *<!-- text -->*). In Figure 3, the date stamps in the leaf labels are page generation times. The difference between the two trees is an added link (i.e., `href`) and a different page generation date stamp.

Using pure Zhang and Shasha, it takes three operations to get from the left-hand tree to the right: two additions for the `<a ...>` node and its *click me* text child, and one substitution for the date stamp. This yields an edit distance of *3*. This distance is larger than one would want, because the page generation time labels are considered unequal, i.e., contribute `+1` to the distance as a substitution. To address this issue we apply a string distance function to text labels. We use *Sift4* [24] for this purpose, which approximates the Levenshtein distance, but is faster. We consider attribute names (e.g., `style=...`) when we compare non-leaf nodes, but not the attribute values. The rationale for this is that variable change in, e.g., form nonces do not create distance. Finally, we normalize the edit distance to be able to compare distances between pairs of trees with different sizes. Using normalization, the distance, D_n, between the trees in Figure 3, with 6 and 8 nodes respectively, becomes:

$$D_n(t_1, t_2) = \frac{2}{6+8} = \frac{1}{7} \quad (1)$$

The resulting $[0..1]$ distance is fed to a binary classifier that considers content similar (i.e., the origin is exposed) or dissimilar, using a straightforward threshold comparison. We explain the threshold selection next.

E. Threshold Selection

We assume that the distance function, D_n, follows two distributions: one with $\mu_s = 0.0$ for similar DOM trees, and the other with $\mu_d = 1.0$ for dissimilar DOM trees. Figure 4 shows the resulting curves. All distances that we calculate form a sample of either curve. Although we can't tell to which curve a calculated distance contributes, the curves intersect and create a mimimum at `t`, which is a reasonable threshold to use in the binary classifier, i.e., $D_n \leq t \implies similar$.

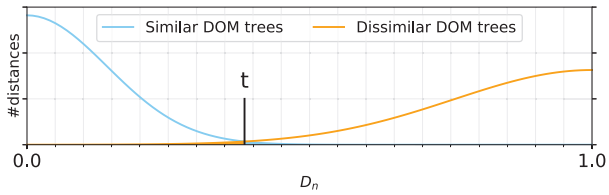

Fig. 4: Distance distributions for similar and dissimilar content

V. Data Sets

We analyze and correlate three data sets in this paper, all of which cover a period of 16 months, from January 22, 2016 to May 22 2017. The first data set is derived by a large-scale, active DNS measurement platform that provides us with DNS measurement data for Web sites (Section V-A). The second data set tracks which Web sites outsource protection to DPS (Section V-B). The third and final data set contains DoS attack events inferred from backscatter to a large network telescope (Section V-C).

A. Active DNS Measurements

To investigate origin exposure among Web sites that use a DPS we rely on the large-scale, active DNS measurement platform: OpenINTEL [25], [26]. The OpenINTEL platform collects daily snapshots of the content of the DNS by structurally querying a set of domain names for their resource records. This includes any RR type tied to the exposure vectors outlined previously, in Section III (e.g., `MX`, `NS`, and `TXT`).

We use a subset of the data that OpenINTEL measures. Specifically, we use measurement data for domain names that belong to Web sites that are on the Alexa's Top 1M (cf. https://www.alexa.com/). The details of this measurement data

start	days	source	Web sites	data points	size
2016-02-22	486	Alexa	3.3M	7.1G	205.0GiB

TABLE I: Active DNS data set for the Alexa Top 1M

provider	Web sites
Akamai	30.6k
CloudFlare	357.2k
DOSarrest	3.2k
F5	5.0k
Incapsula	19.6k
Level 3	13.0k
Neustar	30.0k
Verisign	6.6k

TABLE II: DDoS Protection Service use among Alexa top 1M Web sites for each of the 8 DPS providers that we consider.

are shown in Table I. The total number of Web sites seen over the 16 months period is *3.3*M.[4] The *data points* column shows the total number of *7.1*G collected data points (e.g., `A`, `CNAME`, and `NS` records). The *size* column shows the size of the compressed measurement data using Parquet columnar storage[5], with a total of *205.0*GiB.

The OpenINTEL platform does not target an extensive set of labels by default. That is, it does not brute-force hundreds of 3LDs for any domain it measures. As outlined in Section III-A, however, 3LDs may expose the origin of a Web site. To analyze this exposure vector we instructed OpenINTEL to send out queries for commonly-used labels.[6] The set of labels that we used to this end was provided by SIDN, the `.nl` registry operator, who used their ENTRADA platform [27] to identify the 1000 most-frequently queried labels at authoritative name servers for the `.nl` zone.[7,8]. Among the top-queried labels we find, e.g., `www-origin` and `direct-connect`, which are DPS-specific labels to bypass diversion, as outlined in Section III. Other labels include `mail`, `smtp`, and `ftp`.

B. DDoS Protection Services

We analyze the extent to which Web sites that are long-term, always-on DPS users (i.e., continously divert traffic) expose their origin to direct attacks. Our data set on DPS providers gives us usage information for all Web sites on the Alexa Top one million. This data set was created using a methodology that we previously published in [10]. This methodology relies on the active DNS measurement data outlined in Secion V-A. The DPS use data set covers the use of eight leading DPS providers that support DNS-based traffic diversion [28]. Specifically, these are Akamai, CloudFlare, DOSarrest, F5 Networks, Incapsula, Level3, Neustar, and Verisign. Table II shows the details of the data set in terms of the total number of Web sites that we associate with each of the eight providers, over the period of 16 months. In Section VI we extract long-term, always-on Web sites from the DPS use data set, in accordance with out methodology.

[4]The number is significantly higher than 1 million due to changes in the Web site ranking, especially in the long tail.

[5]https://parquet.io/

[6]As we will show in Section VI, we targeted only domain names of interest to avoid burdening the DNS.

[7]Any query for a 3LD for which a `NOERROR` was returned was counted (once) towards the label's rank, over a one-month period.

[8]To avoid sending queries for labels that are unlikely to have IP address records we first removed labels from the top 1000 that are nonviable (e.g., `_dmarc` & `_domainkey`).

start	days	#events	#targets	#/24s	#/16s	#ASNs
2016-01-22	486	7.95M	1.28M	0.41M	23083	20096

TABLE III: DoS attack events data

C. DoS Attack Events

The third data set contains DoS attack events inferred from backscatter packets that reach the UCSD Network Telescope [29], which is a largely-unused `/8` network that is operated by the University of California San Diego. Network telescopes, also called darknets, passively collect unsolicited traffic sent to routed regions of address space that do not contain any hosts. Unsolicited traffic results from, e.g., scans, misconfigurations, and backscatter from DoS attacks. Any substantial DoS attack that involves uniformly randomly spoofed addresses should be visible on a `/8` darknet, as it receives approximately 1 out of every 256 backscatter packets.

Attacks are inferred by an implementation of the detection and classification methodology described by Moore et al. [30], which identifies randomly spoofed DoS attacks in traffic that reaches the telescope. The implementation is a Corsaro [31] plugin that is openly available [32].

Table III summarizes the data set of attack events. We observe a total of about *8* million attacks over the 16-month period, targeting *1.28*M unique IP addresses in *20*k distinct Autonomous System (AS) numbers.

provider	Web sites	provider	Web sites
Akamai	2100	Incapsula	2854
CloudFlare	4183	Level 3	1173
DOSarrest	245	Neustar	39
F5	265	Verisign	25
total		10884	

TABLE IV: Per DPS the numbers of long-term, always-on Web sites that we analyze further

VI. Results

A. Long-Term, Always-On Customer Web Sites

By our methodology, our analysis starts with a selection of long-term, always-on customers. Table IV shows the number of resulting Web sites for the Alexa Top 1M.[9] We find a total of 10884 long-term, always-on Web sites.

B. Origin IP Address Candidates

We pulled candidate origin IP addresses from the OpenINTEL data set for each of the selected Web sites.[10] For 9260 out of 10884 Web sites (*85.08%*) we found potential origin IP

[9]Given the large always-on customer base of Cloudflare we randomly sampled `1:10` out of 41830 always-on customers.

[10]We filtered 3556 IP addresses in private ranges, 0 invalid addresses, and 1162 routed to a DPS.

addresses. The other 1624 had no potential exposure in their DNS configuration. Table V shows the number of Web sites potentially exposed per vector, and the number of IP addresses found for each of the exposure vectors.[11]

exposure vector	Web sites (%)	#addresses
third-level domain	7928 (85.62%)	*753.6*k
IP address history	3744 (40.43%)	*22.3*k
SPF	2396 (25.87%)	*6.5*k
conflicting record	2314 (24.99%)	*3.1*k
mail exchanger	2091 (22.58%)	*2.8*k
name server	401 (4.33%)	*1.0*k
IPv6	173 (1.87%)	*0.2*k
total	19047	*789.5*k

TABLE V: Number of potentially exposed Web sites per exposure vector along with the total number of IP addresses

C. Scraping Results

The selection of Web sites along with their candidate origin IP addresses were fed to our scraper. For 9170 out of 9260 potentially exposed Web sites we got an answer for the regular HTTP request. That makes *99.03%*.[12] For *96.4%* of 9170 Web sites we got a `HTTP STATUS OK`. Most other responses were `NOT FOUND`, i.e., not the origin (anymore), or `FORBIDDEN`.[13] . We also see various DPS-specific codes such as `523` (*Cloudflare: Origin is unreachable*) and `521` (*Cloudflare: Web server refused connection*).[14]

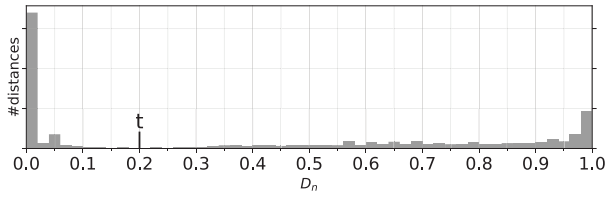

Fig. 5: All pairwise DOM-tree distance calculations

D. Exposed Web Site Origins

We calculated the distances for all DOM-tree pairs in our scraping data to determine the threshold for D_N. Figure 5 shows the results. We find a minimum at $t = 0.2$. Using this threshold, we find that 4408 of the initial, 10884 selected Web sites actually expose the origin (*40.5%*). We will go over statistics for the individual exposure vectors next.

The largest exposure vector are third-level domains. Of the analyzed Web sites, *27.95%* expose their origin IP address on one of the commonly-used labels. That's 3042 out of 10884 Web sites. Table VI shows the five most-common labels on which the origin is exposed. The top two labels (i.e., `direct` and `origin-www`) are inspired by DPS documentation for non-proxiable traffic. The `cpanel` label in the third position is specific to *cPanel*, a control panel for Web sites.[15] The `webmail` label is typically used for Webmail software that is served by a Web server from a document root other than the primary Web site. If domain name administrators forget to divert traffic on this label then it might expose the origin. We checked a few of these cases by hand and encountered instances of Webmail software (e.g., Roundcube[16]).

label	exposes (%)
`direct`	418 (13.7%)
`origin-www`	392 (12.9%)
`cpanel`	387 (12.7%)
`webmail`	381 (12.5%)
`dev`	372 (12.2%)

TABLE VI: Common origin-exposing 3LD labels

label	exposes (%)
`mail`	292 (36.3%)
`mx`	4 (0.5%)
`mx1`	3 (0.3%)

(a) `MX` labels

label	exposes (%)
`ns1`	27 (61.4%)
`ns2`	16 (36.4%)
`ns`	1 (2.3%)

(b) `NS` labels

TABLE VII: Common labels in mail exchanger and name server records that expose the origin of Web sites

Exposure through IP address history comes second. We found the origin of *13.70%* of the analyzed Web sites based on historic IP address data. This comes down the 1491 of 10884 Web sites. It should be noted that for the always-on customers that started using a DPS before the first day of data set, we do not have any historic addresses to pull from the active DNS data. As such, *13.70%* is a lower bound.

By having **conflicting records**, *11.80%* of Web sites expose their origin. Within this exposure type, the majority of Web sites (*94.39%*) have an IP address on the domain root (i.e., the `@`). A small percentage (*5.61%*) have a conflicting IP address on the `www` label. In a handful of cases, Web sites expose the origin on both. Akamai sees most customers that have this type of origin exposure, with *34.34%* of Web sites.

By the **mail exchanger** record, *7.40%* of Web sites expose their origin. Table VIIa shows the three most-common labels in `MX` records that exposure the origin of Web sites. The commonly used `mail` label dominates, which *36.4%* of all Web sites within this exposure type use. We find a lot of labels of the form `dc-<rand>` in `MX` 3LDs. As outlined in Section III, these labels can be traced to CloudFlare. Because these labels are unique and usually occur once, they create a long tail in the ranking of label occurrences. CloudFlare is also the DPS that sees the highest `MX` exposure, at *17.07%*.

Of the analyzed Web sites, *3.22%* expose their origin through the **Sender Policy Framework** information that is published in the DNS. We encountered only IPv4 addresses in SPF records that actually expose the origin. That is, from all IPv6 addresses in SPF information, not a single address exposes a dual-stacked origin.

[11] Note that these are upper bounds on the number of requests required by our scraper, because multiple vectors for a given Web site might overlap on a single address, in which case it needs to be scraped "directly" just once.

[12] The other *0.97%* led to a connection error of some form (e.g., connection, request, or read timeouts).

[13] A `FORBIDDEN` could be given by the origin to requests from anything but the reverse proxy, but given the lack of content we assume no exposure.

[14] "Direct" requests are skipped for Web sites without regular content.

[15] https://cpanel.com/

[16] https://roundcube.net/

	DPS provider								
Exposure Vector	Akamai	CloudFlare	DOSarrest	F5	Incapsula	Level 3	Neustar	Verisign	**All**
third-level domain	38.19%	30.91%	15.10%	12.83%	23.76%	16.37%	10.26%	8.00%	27.95%
IP address history	21.67%	13.20%	3.27%	7.17%	13.70%	5.54%	0.00%	4.00%	13.70%
SPF information	0.81%	5.71%	1.22%	0.38%	2.94%	0.60%	0.00%	0.00%	11.80%
conflicting records	34.43%	6.36%	2.45%	2.26%	4.38%	12.87%	12.82%	8.00%	7.40%
mail exchanger	0.19%	17.07%	0.82%	0.00%	2.87%	0.26%	0.00%	0.00%	3.22%
name server	0.10%	0.26%	0.00%	0.00%	1.02%	0.17%	0.00%	0.00%	0.88%
IPv6 address record	0.33%	0.31%	0.41%	0.00%	2.59%	0.09%	0.00%	0.00%	0.40%
combined	54.05%	46.04%	19.18%	15.85%	34.76%	21.74%	17.95%	16.00%	40.50%

TABLE VIII: Per DPS and per exposure vector the percentages of Web site customers that expose their origin

As for **IPv6 IP address exposure**, of Web sites that are dual-stacked (i.e., those that have an `AAAA` record as well as an `A` record), only *0.88%* expose the origin on the IPv6 address. Among DPS providers, for Incapsula we see that most Web sites expose their origin this way, specifically *2.59%*.

Finally, *0.4%* of Web sites expose the origin through a **name server** record. That is, the authoritative name server for the domain name of the Web site runs on the origin Web server. Table VIIb shows the three most-common labels in `NS` records that expose the origin. The largest exposure of this type among DPS providers is for Incapsula, where *1.00%* of the Web sites expose the origin through a `NS` record.

Table VIII summarizes statistics per exposure vector, along with a per-DPS breakdown. It should be noted that for some providers the number of always-on Web sites is low (i.e., of the order of tens to only a few hundred), which might make the breakdown for those providers less representative.

We investigated the IP addresses on which Web sites are exposed and found several addresses that are shared by multiple (exposed) Web sites. That is, even though Web sites are individually exposed through one of the exposure vectors, they end up on a shared IP. One reason for this is that Web sites share the same owner.[17] Another reason is that Web sites are placed at a party that provides hosting (e.g., Google or Amazon).[18] The common-most AS numbers associated with IP addresses on which Web sites are exposed belong to: Amazon, OVH, UnifiedLayer, Hetzner, and DigitalOcean.

E. Attacked Web Sites

We correlated the set of Web sites with verified origin exposure with our data set of attacks and found that the origin of 843 of 4408 Web sites was attacked after the Web site had started using a DPS. This comes down to *19%* of exposed Web sites. Given that multiple Web sites (or even non Web services) can be hosted on the same IP address, we cannot definitively say if the Web site was the target of the attack.

Our attacks data has per event an intensity attribute, expressed in terms of maximum packets per second (`pps` average) that the victim backscatters to the network telescope. This value ranges up to 310k. A higher intensity indicates a more powerful attack. The top 1 percentile of all attacks sees a `pps` of 200 or up, which equals an attack traffic rate of 51200 (the number should be multiplied by 256 as the telescope is a `/8`). 205 of exposed Web sites see an attack of this strength on their origin, showing that strong attacks are launched on exposed origins.

[17] As an example, we found adult content Web sites that were seemingly related from similar names and which share hosting.

[18] We have seen one case where as much as 22 Web sites can be traced to the same IP address from Amazon.

VII. Related Work

In 2013, McDonald [7] brought attention to the fact that content delivery security (e.g., network traffic diversion to DDoS Protection Services) can in some cases be bypassed, leaving Web sites vulnerable to attacks. Later in 2013, Nixon and Camejo [8] attracted even more attention to this fact. In 2015, Vissers et al. [9] performed the first study into origin exposure at scale, for five DPS providers. Our study extends all the aforementioned work. First, we add more exposure vectors. Second, we look at an extensive set of nine DPS providers, for which our previous work shows a clear trend in adoption [10]. Third and final, we use a data set of DoS attack events inferred from backscatter traffic to a large network telescope to determine if the origin IP addresses of exposed Web sites are attacked.

VIII. Conclusions

We studied the extent to which Web sites that outsource protection to DDoS Protection Services (DPS) by means of DNS-based network traffic diversion expose their origin IP address through DNS misconfiguration. Our study covers Web sites on Alexa's Top 1 million list that are long-term customers of eight leading protection services. We evaluated previously known as well as novel exposure vectors to find the potential origin IP addresses of Web sites at scale. By using a DOM-tree comparison method we were able to verify that *40.5%* of the studied Web sites expose a (responsive) origin, which means that the protection of these Web sites can be bypassed, and that these Web sites are vulnerable to direct DoS attacks. As a consequence, the use of a DPS might not be as safe as most Web site operators expect. Our results reaffirm previously known errors, and identify novel DNS misconfigurations, all of which operators should address to fortify their defenses.

We also correlated exposed origin IP addresses with a large data set of DoS attacks. Our results show that the origin of *19%* of Web sites see (high-intensity) attacks after protection was outsourced, which indicates that correct management and configuration are needed to ensure the efficacy of DPS use.

ACKNOWLEDGMENTS

This work is part of the NWO: D3 project, which is funded by the Netherlands Organization for Scientific Research (628.001.018). This research was made possible by OpenINTEL, a joint project of the University of Twente, SURFnet, and SIDN. We thank the anonymous reviewers and Alberto Dainotti for their valuable feedback.

REFERENCES

[1] P. Paganini, "The hosting provider OVH continues to face massive DDoS attacks launched by a botnet composed at least of 150000 IoT devices." http://securityaffairs.co/wordpress/51726/cyber-crime/ovh-hit-botnet-iot.html, September 2016.

[2] G. C. Moura, R. d. O. Schmidt, J. Heidemann, W. B. de Vries, M. Muller, L. Wei, and C. Hesselman, "Anycast vs. DDoS: Evaluating the November 2015 Root DNS Event," in *Proceedings of the 2016 ACM Internet Measurement Conference (IMC'16)*, 2016.

[3] S. Hilton, "Dyn Analysis Summary Of Friday October 21 Attack," http://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/, October 2016.

[4] J. Pescatore, "DDoS Attacks Advancing and Enduring: A SANS Survey," SANS, 2014.

[5] E. Nygren, R. K. Sitaraman, and J. Sun, "The Akamai Network: A Platform for High-performance Internet Applications," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 3, pp. 2–19, 2010.

[6] C. Huang, A. Wang, J. Li, and K. W. Ross, "Measuring and Evaluating Large-Scale CDNs," in *Microsoft Research Technical Report MSR-TR-2008-106*, October 2008, (full paper withdrawn from the 8th ACM SIGCOMM Conference on Internet Measurement (IMC08)).

[7] D. McDonald, "The Pentesters Guide to Akamai," 2013.

[8] A. Nixon and C. Camejo, "DDoS Protection Bypass Techniques," Black Hat USA, 2013.

[9] T. Vissers, T. van Goethem, W. Joosen, and N. Nikiforakis, "Maneuvering Around Clouds: Bypassing Cloud-based Security Providers," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1530–1541.

[10] M. Jonker, A. Sperotto, R. van Rijswijk-Deij, R. Sadre, and A. Pras, "Measuring the Adoption of DDoS Protection Services," in *Proceedings of the 2016 ACM Internet Measurement Conference (IMC'16)*, 2016, pp. 279–285.

[11] C. Rossow, "Amplification Hell: Revisiting Network Protocols for DDoS Abuse," in *NDSS*, 2014.

[12] M. Sargent, J. Kristoff, V. Paxson, and M. Allman, "On the Potential Abuse of IGMP," *ACM SIGCOMM Computer Communication Review*, vol. 47, no. 1, 2017.

[13] J. Czyz, M. Kallitsis, M. Gharaibeh, C. Papadopoulos, M. Bailey, and M. Karir, "Taming the 800 Pound Gorilla: The Rise and Decline of NTP DDoS Attacks," in *Proceedings of the 2014 ACM Internet Measurement Conference (IMC'14)*, 2014, pp. 435–448.

[14] R. van Rijswijk-Deij, A. Sperotto, and A. Pras, "DNSSEC and its potential for DDoS attacks: a comprehensive measurement study," in *Proceedings of the 2014 ACM Internet Measurement Conference (IMC'14)*, 2014, pp. 449–460.

[15] J. Mirkovic, S. Dietrich, D. Dittrich, and P. Reiher, *Internet Denial of Service: Attack and Defense Mechanisms (Radia Perlman Computer Networking and Security)*, 2004.

[16] "A Complete DDoS Protection Solution From a Leading Provider of Internet Infrastructure," https://www.verisign.com/en_US/security-services/ddos-protection/index.xhtml, accessed: 2017-05-24.

[17] J. Liu, "Firewall Rule: Global Traffic Management - SiteShield ACL," https://community.akamai.com/community/cloud-security/blog/2017/04/21/firewall-rule-global-traffic-management-siteshield-acl, 2017, accessed: 2017-05-24.

[18] S. Rose, R. Chandramouli, and A. Nakassis, "Information leakage through the domain name system," in *Proceedings of the Cybersecurity Applications and Technologies Conference for Homeland Security (CATCH'09)*, 2009, pp. 16–21.

[19] F. Weimer, "Passive DNS replication," in *FIRST Conference on Computer Security Incident*, 2005.

[20] S. Kitterman, "Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1," RFC 7208 (Internet Standard), Internet Engineering Task Force, April 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7208.txt

[21] N. Sullivan, "DDoS Prevention: Protecting The Origin," https://blog.cloudflare.com/ddos-prevention-protecting-the-origin/, 2013, accessed: 2017-05-24.

[22] K. Zhang and D. Shasha, "Simple Fast Algorithms for the Editing Distance between Trees and Related Problems," *SIAM Journal on Computing*, vol. 18, no. 6, pp. 1245–1262, 1989.

[23] A. P. Rosiello, E. Kirda, C. Kruegel, and F. Ferrandi, "A Layout-Similarity-Based Approach for Detecting Phishing Pages," in *Proceedings of the 3rd International Conference on Security and Privacy in Communications Networks and the Workshops (SecureComm'07)*, 2007, pp. 454–463.

[24] "Super Fast and Accurate string distance algorithm: Sift4," 2014, https://siderite.blogspot.com/2014/11/super-fast-and-accurate-string-distance.html.

[25] R. van Rijswijk-Deij, M. Jonker, A. Sperotto, and A. Pras, "A High-Performance, Scalable Infrastructure for Large-Scale Active DNS Measurements," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 34, no. 6, pp. 1877–1888, 2016.

[26] "OpenINTEL Active DNS Measurement Project," 2015, http://www.openintel.nl/.

[27] M. Wullink, G. C. Moura, M. Müller, and C. Hesselman, "ENTRADA: A high-performance network traffic data streaming warehouse," in *Proceedings of the 2016 IEEE/IFIP Network Operations and Management Symposium (NOMS'16)*, 2016, pp. 913–918.

[28] R. Holland and E. Ferrara, "The Forrester Wave™: DDoS Services Providers (Q3 2015)," Forrester Research, Inc., July 2015.

[29] "The CAIDA UCSD Near-Real-Time Network Telescope – 2016-01 – 2017-05," http://www.caida.org/data/passive/telescope-near-real-time_dataset.xml.

[30] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring Internet Denial-of-service Activity," *ACM Transactions on Computer Systems*, vol. 24, no. 2, pp. 115–139, 2006.

[31] A. King, "Corsaro," 2012, http://www.caida.org/tools/measurement/corsaro/.

[32] ——, "Corsaro RS DoS Plugin," 2012, https://www.caida.org/tools/measurement/corsaro/docs/plugins.html#plugins_dos.