The Best Bang for the Byte: Characterizing the Potential of DNS Amplification Attacks

Douglas C. MacFarland[a], Craig A. Shue[a,*], Andrew J. Kalafut[b]

[a] *Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA, USA*
[b] *Grand Valley State University, 1 Campus Drive, Allendale, MI, USA*

Abstract

DNS amplification has been instrumental in over 34% of high-volume network DDoS attacks, with some floods exceeding 300Gbps. Today's best practices require Internet-wide cooperation and have been unable to prevent these attacks. In this work, we investigate whether these best practices can eliminate DNS amplification attacks and characterize what threats remain. In particular, we study roughly 130 million DNS domains and their associated servers to determine the DNS amplification potential associated with each. We find attackers can easily use these servers to create crippling floods and that few servers employ any protection measures to deter attackers.

Keywords: Domain Name System, Denial of Service Attacks, Internet Measurements

1. Introduction

From an offensive standpoint, DNS amplification attacks have two attractive qualities: 1) they mask the identity of the attacking systems and 2) they conscript innocent bystanders into increasing the damage associated with the attack. In a July 2016 attack, a DDoS attack using DNS amplification led to a flood of over 363Gbps [1] while a recent Akamai study showed over 400 DDoS attacks using DNS amplification from November 2015 through February

*Corresponding author.
Email address: cshue@cs.wpi.edu (Craig A. Shue)

2016 [2].

To effectively launch a DNS amplification attack, the attacking machine must be able to "spoof" its source IP address in a DNS lookup packet. The attacker will select a victim and create a DNS query. However, rather than specifying the attacker's own IP address as the source of the packet, the attacker supplies the victim's IP address as the source. The attacker then sends the DNS query packet on to an innocent third-party DNS server. That DNS server, unaware of the address forgery, dutifully replies to the DNS query, sending a response to the victim. The victim must then recognize that the DNS response is unneeded and then discard it. Even worse, the attacker can carefully select DNS queries that are small in size (e.g., 75 bytes) that will cause the DNS server to send large responses (e.g., 1500 bytes) to the victim. This can allow an attacker to expend relatively little bandwidth to create a large flood at the victim.

A DNS amplification attack puts the victim in a rough spot: the victim must inspect DNS queries and responses to determine whether a response is legitimate or not. Even if the victim can do so quickly, the attack packets can saturate the victim's upstream network connection. Further, a given reflecting server may happen to offer DNS records that the victim's users legitimately want to access. Simply blocking the IP addresses of all the reflecting servers may cause replies from legitimate queries made by the victim's users to be discarded, causing collateral damage. Even worse, for victims that are connected via cellular networks, such floods could dramatically impact the portions of the cellular network and degrade performance for unrelated network users [3].

Unfortunately, there is little a potential victim organization can do to protect its own network. The best current guidance to prevent DNS amplification attacks require Internet-wide cooperation to lessen the risk of attacks. The United States Computer Emergency Response Team (US-CERT) made a few recommendations [4]: 1) reduce the number of open DNS resolvers, 2) disable public recursion on authoritative DNS servers, 3) rate limit responses [5], and 4) limit IP address spoofing. While the last recommendation, of eliminating IP spoofing, has been recommended for over a decade, over 25% of Autonomous

Systems still allow arbitrary IP spoofing on the Internet [6].

In this work, we investigate two research questions: *What is the attack potential associated with DNS amplification attacks? Would the current recommendations eliminate the attack?*

While eliminating IP spoofing would stop DNS reflection attacks, ensuring universal adoption of those measures has been elusive. We investigate whether attackers can still launch damaging attacks even if all open DNS resolvers are removed and recursion is disabled at authoritative DNS servers. Further, other DNS best practices, such as separating authoritative and recursive DNS servers, DNSSEC, and authenticated DNS queries, have little impact on DNS reflection or amplification attacks. These attacks are possible because the authoritative DNS server replies to public queries and that functionality is inherent to the authoritative server's role in DNS.

In this work, we make the following contributions:

- **Determine the Amplification Risk Associated with Authoritative Servers:** With over 130 million DNS domains registered across 9 top-level domains (TLDs), attackers can issue a large number of unique queries that will be reflected back at victims. We perform DNS queries to each of these domains to determine which queries have the highest amplification factor, or the "biggest bang for the byte." We found that over the last two years, the amplification rate has increased, allowing attackers to create an flood of roughly 1,799 MBytes while only having to transmit 44 MBytes.

- **Determine the Adoption of Resource Record Rate-Limiting:** We queried each of the roughly 1 million unique DNS servers in our study to determine whether they used rate-limiting. For each server, we picked a domain served by the server and issued a query for that domain 30 times in rapid succession to determine whether the server rate limited the responses. We found that 10.23% of servers employed the protective measure, indicating the approach is not widely used in practice.

The rest of this document is structured as follows. In Section 2, we provide

background and survey related work. We describe our measurement methodology in Section 3 and detail the impact of record rate limiting in Section 4. We discuss our findings and impact in Section 5 and conclude in Section 6.

2. Background and Related Work

Traditional reflection attacks, such as the Smurf attack [7], simply forge the source IP address of a packet to be the address of the intended victim. The attacker sends the packet to an innocent third-party system called a *reflector*. The reflector then issues a legitimate reply that arrives at the victim. When a large number of attack packets are sent to reflectors, or when a reflector is a broadcast network address for many hosts, the combined volume at the victim can be crippling.

In a 2001 article, Paxson [8] described how reflectors can be used as part of a distributed reflector denial of service (DRDoS) attack. He argued for five possible defenses against the attacks: 1) filter reflected attack traffic at the victim, 2) prevent source address spoofing, 3) detect and block spoofed packets at the reflector, 4) allow traceback to the origin even through the reflector, and 5) detect the attack traffic from the compromised systems. With the exception of the first defense, in which the victim employs filtering, each of these defenses requires a third-party organization to detect and block attack traffic. The specific third-party organization affected depends on the details of the attack (e.g., the origin of the attack and the particular reflectors in use), but each of them must implement the solution. Solutions which require 100% adoption by third-parties are unlikely to succeed, especially when these third-parties have no incentives for adoption. For example, the second option, source address filtering, is comparatively straightforward for organizations to employ, yet over 25% of Autonomous Systems still allow arbitrary IP spoofing on the Internet [6].

Attackers often try to increase the amount of traffic generated by an attack by having another system involved. These attacks, called *amplification attacks*, typically leverage protocol-specific attributes to increase the attack vol-

ume. Recent attacks using NTP amplification [9, 10] were able to create floods of 400Gbps against a victim. In the NTP attack, the attacker found a list of susceptible NTP servers and, spoofing the IP address of the victim, issued a query requesting a list of the last 600 clients that accessed the server. These NTP responses were much larger than the query, creating a massive amplification attack against the victim. Rossow [11] examined 14 different network protocols to look for reflection attacks that yield significant amplification. Rossow's analysis included DNS, but it was not as comprehensive as our own; their study included only 255,819 authoritative DNS servers from a web crawl while ours evaluates the authoritative servers for over 129 million domains, due to our use of the underlying zone files for several zones. Additionally we did not prefilter based on the deployment of DNSSEC, reducing potential sources of bias. Kührer *et al.* discuss the prevalence of DNS amplifiers, compared to other UDP-based protocols, and discuss fingerprinting techniques [12]; however, they do not expand on the amplification results. The solutions they propose focus on efficient identification, the notification of vulnerable amplifiers for various protocols, and on curtailing ASes that allow spoofing. Finally, Krämer *et al.* [13] explored the role of open DNS resolvers and defenses against those systems.

The most closely related work is our own prior work [14], in which we performed an earlier version of this study. In this work, we have performed two additional data collection snapshots, queried for additional hosts in each domain, and expanded our analysis of rate limiting. Other recent work has examined the impact that DNSSEC can have in DNS amplification attacks [15]. That work performs some similar measurements, although not as extensive, but does not include an evaluation of DNS rate limiting.

US-CERT recommends that organizations focus on eliminating open DNS resolvers [4], which echoes RFC 5358 [16]. However, this advice ignores the hundreds of thousands of authoritative DNS servers that are, by design, required to answer DNS queries to anyone who asks. These servers are well provisioned and capable of handling large volumes of traffic [17]. Attackers could use these servers to launch crippling attacks, even without using open resolvers. Accord-

ingly, we focus on the risks associated with authoritative servers in this work.

Other reflector and amplification attacks can be damaging. However, we focus on DNS amplification because the protocol is widely used and the amplification attack can be indistinguishable from legitimate usage. Further, measures such as filtering, which may be used to mitigate other amplification attacks, would have unacceptable consequences for DNS (such as leaving a victim without the ability to resolve host names).

3. DNS Amplification Potential

We begin by discussing attacker options in launching DNS amplification attacks and then describe our data collection and the results of our measurements.

3.1. The Biggest Bang for the Byte

DDoS attackers want to launch the most effective flood with the lowest cost. As a result, attackers may be inclined to issue DNS queries that will yield the greatest amplification factor to minimize the bandwidth cost for the attacker while maximizing the traffic at the victim.

To optimize for amplification, attackers have several variables they can consider. The first is the type of query to issue. Our prior work [18] shows that `A` records, which provide the IPv4 address for an indicated host name, are quite common in DNS domains. These queries are often issued by hosts on the Internet, making it hard for DNS server operators to distinguish attack traffic from legitimate queries. Some recent DNS amplification attacks have used the `ANY` record type in their queries. The `ANY` query is unique in that it asks the server to supply all DNS resource records associated with the requested host name. Since most DNS clients are requesting specific record types, the `ANY` record is not commonly used and administrators may block or record its use. Snort, for example, could use rule signature that alerts on the use of `ANY` queries as being part of an attack. Some attackers may prefer to use `A` record queries to evade detection while others may embrace `ANY` queries for the potential amplification gain.

Next, attackers must consider the maximum packet size they expect in the DNS server's response. Traditional DNS packets are limited to a maximum length of 512 bytes at the application layer. If the expected DNS response is 512 bytes or less, the attacker can issue a standard query. However, if the expected response size is larger, the attacker can use the extension mechanisms for DNS (EDNS) [19] to tell the DNS server that it is allowed to reply with larger DNS packets. To do so, the attacker must include a pseudo-resource record, `OPT`, that indicates the supported packet size. That same `OPT` record can also declare DNSSEC support [20], indicating that the server should send any associated DNSSEC records, which can boost attack response size. The `OPT` record is 11 bytes in size, so the DNS response must grow significantly in order to outweigh the attacker's increased expenditure in bandwidth.

Given all these considerations, a sophisticated attacker may wish to perform a detailed study of the available DNS authoritative servers to determine the response size associated with all the possible queries. From there, the adversary could optimize the queries issued to maximize the amount of traffic at the victim while minimizing the cost in number and size of queries. Accordingly, we perform such a study to determine the risks associated with the most sophisticated DNS amplification attacks that only use authoritative DNS servers.

3.2. Data Collection

We begin our data collection by obtaining a list of authoritative DNS servers and the zones they serve. We contacted the zone maintainers for nine generic top-level domains (gTLDs): `biz`, `com`, `info`, `mobi`, `name`, `net`, `org`, `travel`, and `us`. We obtained zone files providing the host names and IP addresses of the name servers associated with each domain registered under these gTLDs. We obtained this data on three separate occasions: in July 2013, January 2015, and May 2015. The total number of unique domains in these snapshots ranged from roughly 129 million to 136 million. Each domain may designate multiple authoritative name servers for the domain. Further, some name servers host many different domains. In our snapshots, we found that roughly 1 million

Data Set Label	Zone File Date	Probe Dates	Unique Domains	Unique NS IPs	Unique Domain-NS Pairs
July 2013	Jul. 3, 2013	Jul. 29 to Aug. 1, 2013	129,300,870	1,101,446	363,263,970
January 2015	Jan. 2, 2015	Jan. 4 to Jan. 10, 2015	136,178,466	1,058,859	379,960,483
May 2015	May 18, 2015	May 20 to May 26, 2015	134,783,222	1,076,345	399,997,897

Table 1: Statistics about each of the three DNS probing trials.

unique name server IP addresses were present across all the domains studied. We describe each of these snapshots at a high level in Table 1.

Once we obtained the name servers and associated domains, we could begin the study. To explore all the variables an attacker must consider, we issued multiple queries for each domain. For each domain, we issued `A` record and `ANY` record queries. For each type of query, we queried both with `EDNS` and DNSSEC support enabled and without. We also explored queries for IPv6 records, the `AAAA` record, but they were not widely used and did not provide a meaningful amplification over the other queries types. Accordingly, we omit any further discussion of these records.

For all the domains, we queried for the domain itself (e.g., `example.com`) in all three trials. Additionally, in the January 2015 and the May 2015 trials, we also queried for the `www` host in each of the respective domains (i.e., we concatenated the string `www.` and the domain name to form a host name, such as `www.example.com`), which is a common host name used for Web servers. This allowed us to compare the amplification potential of a specific host in the domain as compared to the domain as a whole.

For each snapshot, we had to perform billions of queries. We queried for each domain twelve times each to test possible combination of record type (`A`, `ANY`, `AAAA`), EDNS and DNSSEC support status (enabled or disabled), and queries for the domain itself or the `www` host in each domain. Since these queries follow Internet standards, we believe they pose little risk for the servers being examined. We did allow the operators of the DNS servers to opt-out of the study; however, no operators contacted us to do so.

From a practical standpoint, we used a dedicated querying process and a

separate packet capture process to collect and store each of the DNS responses sent to our server. Our querying process included a number of script instances running in parallel issuing the appropriate DNS query to the servers. However, these querying scripts did not process any responses. A separate packet capturing process recorded the results of the queries. We correlated the query time with the query response. Some query or response packets may have been dropped in transit, but for expediency, we accepted these losses and did not attempt a retransmission. Accordingly, each of the results we report will be conservative estimates of possible amplification.

3.3. Analysis of Servers and DNS Responses

We now examine the DNS responses received from the queries described in the previous subsection. We exclude malformed packets from the analysis since we cannot properly parse them. Such packets amount to no more than 0.25% of the DNS responses in each data set.

We show the overall success rates of our queries in Figure 1, and the overall amplification ratio that results from the responses we received, in Figure 2. In our calculation of amplification ratio, we used application layer packet sizes only (i.e., the DNS headers and payload). We exclude the datalink, network, and transport layer headers from our analysis and focus on the potential of DNS. While we could perform a similar calculation for ratios including these headers, the main impact is a slightly lower amplification rate since those headers in the query and response are usually the same size.

Across all query types and data sets, using DNS reflection more than doubles the application layer traffic volume. When using EDNS, we see lower amplification ratios in each data set than we observed when not using it. The distribution of amplification ratios for all four query types in the July 2013 data set is shown in Figure 3. We observe that except at the upper end of the distribution, the effect of using EDNS appears to be a slight shift in the distribution towards lower amplifications. This shift is due to the large amount of cases where EDNS is providing overhead by increasing the size of the query (11 bytes are needed to

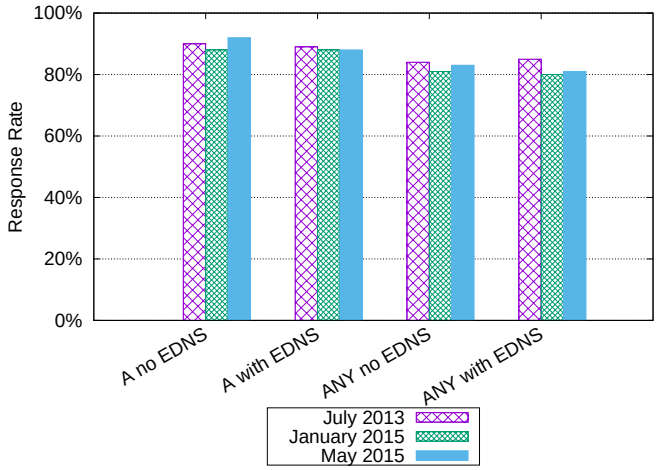

Figure 1: Response rate to DNS queries for domain names in all trials.

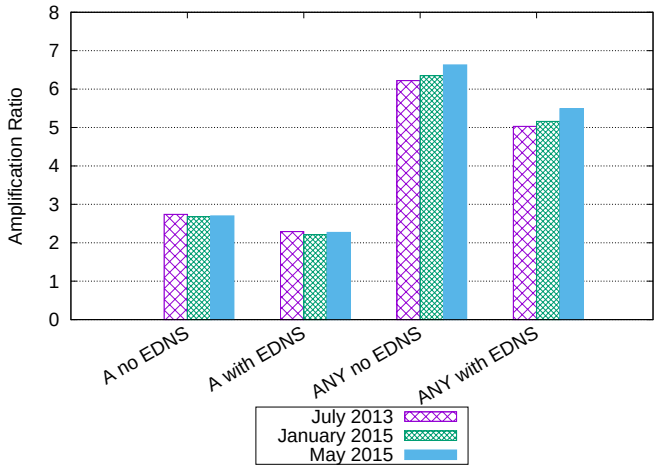

Figure 2: Observed overall amplification ratios for domain name queries in all trials.

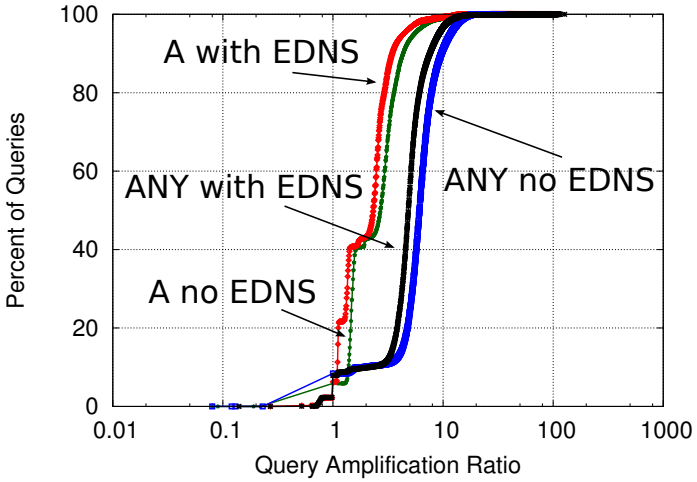

Figure 3: Cumulative distribution function of the amplification ratio compared to the percent of queries for each data set in 2013 trial.

add an `OPT` record to enable EDNS), but not providing any actual change in the size of the response. Specifically, we observe that with EDNS enabled, depending on the specific data set, only 0.27% - 0.35% of `A` record queries and 1.40% - 4.59% of `ANY` queries produced responses larger than the 512 bytes allowed without EDNS. Simply put, an attacker does not benefit from using EDNS in most cases since few responses must be shortened to fit within 512 bytes.

To provide context for these results, we consider the theoretical maximum amplification, at the application layer, for DNS with EDNS using the recommended maximum response size of 4096 bytes. The DNS header itself is 12 bytes, with an additional $n+5$ bytes for a query record, with a domain name of length n, and another additional 11 bytes for the OPT record to enable EDNS. The average maximum amplification with EDNS can then be expressed as $\frac{4096}{N+28}$ where N is the average domain name length in the queries. In our dataset, the average domain name length was 17 characters, which yields a maximum average amplification of roughly 91.02. Our overall amplifications are much lower than this, indicating most queried systems are not providing maximum-sized responses.

Query		Bytes Sent (MB)			Bytes Received (MB)			Overall Amplification Ratio		
Type	EDNS?	7/2013	1/2015	5/2015	7/2013	1/2015	5/2015	7/2013	1/2015	5/2015
`A`	no	34	31	31	485	439	448	14.42	14.23	14.56
`A`	yes	44	43	43	725	728	805	16.37	17.13	18.99
`ANY`	no	35	26	26	534	462	465	15.32	17.33	18.04
`ANY`	yes	44	43	44	1,444	1,701	1,799	32.77	39.46	41.49

Table 2: DNS Responses to Queries for Domain Name for Top 1 Million Largest Responses in Each Trial.

The degree of amplification presented in the results so far is the overall ratio. This is the ratio that an attacker could expect when choosing a large number of domains randomly for reflection attacks. That random selection would be low effort for the attacker, and still achieves some amplification. However, much better amplification ratios can be achieved by an attacker focusing on just the domains yielding the largest DNS responses, instead of all domains. Such domains can be determined by an attacker in advance of any use in an attack. Table 2 and Figure 4 show statistics on the amplification achieved by focusing on the top one million largest DNS responses in each data set. These packets make up roughly 0.25% to 0.3% of each data set, which consists of the 363-400 million queries (one query for each unique domain-server pairs in Table 1).

While EDNS did significantly help an attacker sending queries to random domains, it is more beneficial for an attacker who focuses on those providing the most amplification. In all groups, EDNS yielded a notable increase in amplification among the million largest amplifying responses. This selective querying can help an attacker increase the amplification ratio to over 14.23 in the case of `A` records without EDNS and up to 41.49 in the case of `ANY` queries with EDNS enabled. In other words, the May 2015 EDNS ANY queries for the top 1 million responders show an attacker could send 44 MBytes and create 1,799 MBytes of attack traffic.

The attacker receives the best amplification while using `ANY` queries, but we note that this record type may raise suspicions, as there are few known

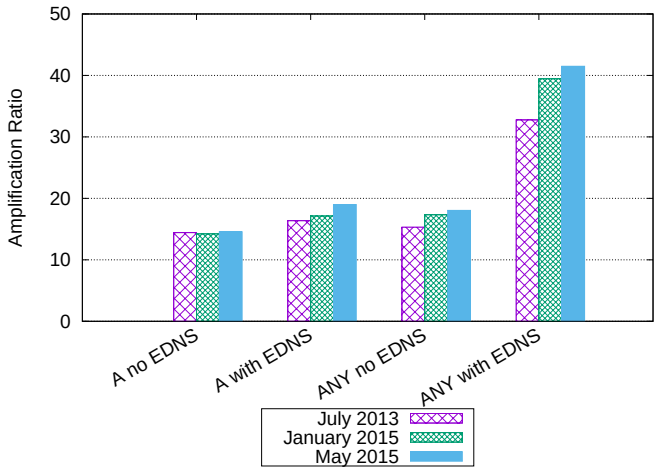

Figure 4: Observed amplification ratios for domain name queries for top 1 million largest responses in all trials.

reasons for legitimate applications to generate a DNS query with this type. An attacker that wishes to use `A` record queries to avoid detection can still achieve an amplification factor of 18.99. As an anecdotal result, in issuing the roughly 1.5 billion DNS queries for each trial associated with this study, our organization was contacted only twice by a queried organization. Organizations may begin filtering `ANY` queries to reduce the amplification factor (such filtering is evident in the lower response rate for our `ANY` queries), but the amplification potential of `A` queries is unlikely to change.

Attackers wanting to maximize amplification will likely use EDNS for the larger possible packet sizes, and `ANY` queries because of the greater number of records returned, which serves to increase actual packet sizes. Our data has shown that this combination produces the highest amplification ratios. Figure 5 shows the distribution of amplification ratios for this data point in each of our three data sets, for the top million largest responses. From this figure we observe that by reducing the number of domains used as reflectors, the attacker can significantly increase the overall amplification factor. The average amplification factor for the top 100,000 largest responses in each data set ranged from 49.9

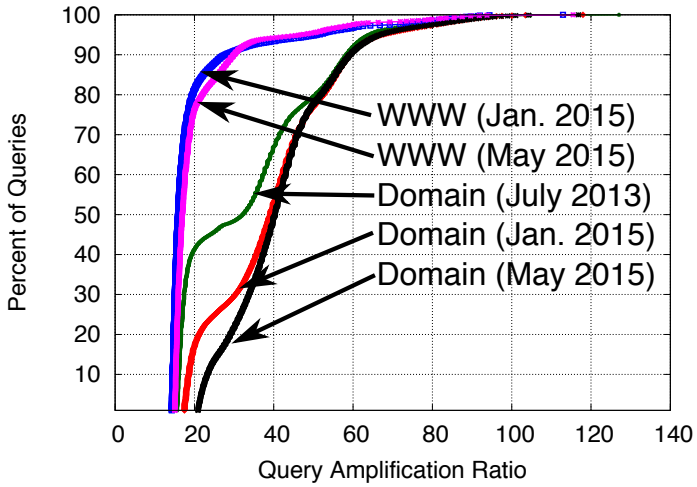

Figure 5: Cumulative distribution function of the amplification ratio compared to the percent of queries for the top 1 million amplified ANY queries with EDNS enabled.

(in the May 2015 WWW sample) to 77.7 (in the May 2015 domain sample).

While attackers want to maximize the amplification factors associated with attacks, they must also ensure they use a large, distributed base of reflectors. If the attackers focus on a small number of highly amplifying reflectors, the reflector bandwidth may become a bottleneck. Even worse, the defenders may be able to filter a small number of reflector IP addresses with little collateral damage. Alternatively, with a small number of reflecting servers being used, the operators of the reflecting servers may be able to detect and filter the attack. To highlight this point, we note that although we received responses from 669,090 (in July 2013) reflecting name servers, a much smaller pool of servers are responsible for the 1 million highest amplifying queries. For the top 1 million `A` record queries, the number of servers ranges from 24,782 in the "without EDNS" group to 24,841 servers in the "with EDNS" group in the 2013 data set. For the top 1 million `ANY` queries, the number of servers ranges from 22,508 in the "without EDNS" group to 28,101 in the "with EDNS" group in the 2013 data set. In other words, less than 3.8% of authoritative name servers are associated with the highest degrees of amplification. In Figure 6, we demonstrate

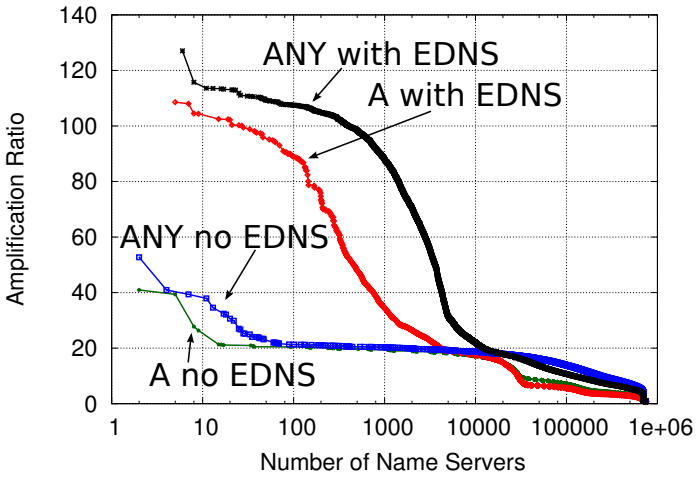

Figure 6: Amplification ratios ordered from the most amplifying server to the least in 2013 trial. Some data points are aggregated for readability.

Query Type		Bytes Sent (MB)		Bytes Received (MB)		Overall Amplification Ratio	
Record	Uses EDNS	Jan. 2015	May 2015	Jan. 2015	May 2015	Jan. 2015	May 2015
`A`	no	35	35	456	461	13.09	13.27
`A`	yes	47	47	851	824	18.17	17.72
`ANY`	no	32	32	464	465	14.55	14.68
`ANY`	yes	45	45	839	875	18.65	19.42

Table 3: DNS Responses to Queries for `WWW` Host for Top 1 Million Largest Responses in Each Trial.

the amplification ratios associated with each name server in the 2013 data set. This pattern continued in our 2015 data sets.

3.4. Effect of Querying for Specific Hosts

Next, we examined the impact of querying for specific host names within a domain instead of for the domain itself. Specifically, we chose to query for the `www` host within each domain because of its widespread use. We show these results for all queries in Figures 7 and 8 and the top 1 million most amplifying queries in Table 3 and Figure 9.

We notice significantly less amplification when querying with the `ANY` type for the `www` host name, as compared to the `ANY` type query for the domain itself.

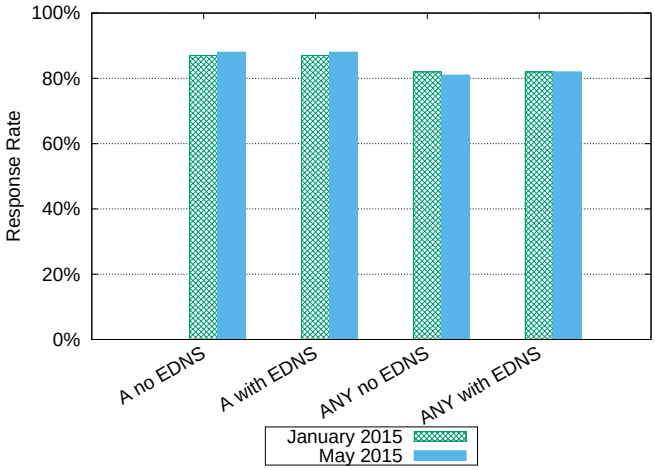

Figure 7: Response rate to DNS queries for WWW host in all trials.

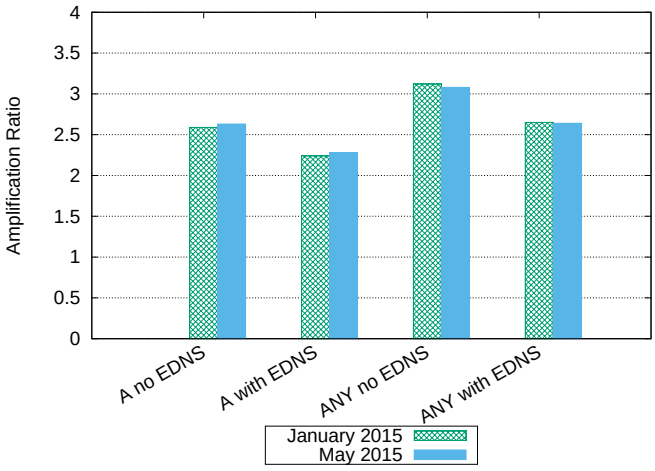

Figure 8: Observed overall amplification ratios for WWW host queries in all trials.

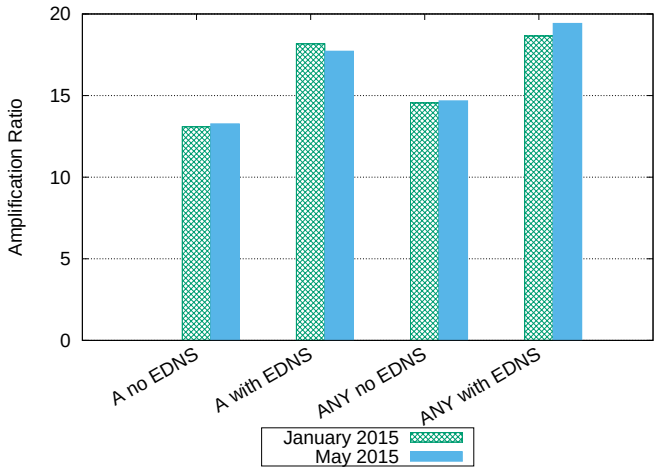

Figure 9: Observed amplification ratios for `WWW` host queries for top 1 million largest responses in all trials.

These results are somewhat intuitive: the `ANY` record for a domain may pick up `MX`, `TXT`, and DNSSEC-related records for the domain as a whole, but those records would not be solicited by an `ANY` request for a specific host within the domain.

The results of `A` record queries on the `www` host are roughly similar to the results of querying for the `A` records of the domain itself. Without EDNS, they are slightly lower in both of our data sets. With EDNS, slightly higher amplification was observed in the January data set, and slightly lower in the May data set. Comparing the amplification ratios for individual domains that responded to both the `www` query and the domain query, we observe that 90.74% to 94.58% of domains had better amplification when using the `www` host name for `A` record queries across the 2015 data sets. However, when querying for `ANY` records, the `www` query was larger in only 25.20% to 26.16% of cases. An attacker using only `A` records might query for both the domain and specific well-known hosts within a domain, such as `www`, since similar levels of amplification are achieved. However, an attacker using `ANY` records would gain the most amplification by focusing on queries to the domain itself.

	Packet Bytes (Percent)		Packet Occurrence %	
Record Type	A	ANY	A	ANY
A	156 (18.19%)	60 (3.22%)	92.1%	96.0%
AAAA	155 (16.87%)	99 (5.82%)	63.6%	30.8%
NS	213 (24.86%)	76 (4.06%)	91.9%	99.5%
MX	-	52 (2.68%)	-	77.9%
SOA	70 (10.27%)	61 (3.26%)	7.9%	96.5%
TXT	-	76 (3.96%)	-	46.6%
RRSIG	706 (71.93%)	1,336 (67.57%)	48.8%	90.9%
DNSKEY	-	412 (19.61%)	-	72.0%
NSEC3	91 (13.44%)	-	7.9%	-

Table 4: Average number of bytes by resource record type for Top 1 million EDNS groups (May 2015, domain query), as well as the occurrence percentages. We omit negligible results for readability.

3.5. Impact of Record Type on Response Size

In Table 4, we show the contributions each resource record makes to the typical DNS packet from the Top 1 million EDNS groups in the most recent trial. In columns 2 and 3 of the table, we show how many bytes, on average, the record constitutes in each packet in which the record appears, along with the overall percentage of the response packet that it constitutes. However, no record occurs in every single packet. In columns 4 and 5, we show how often these records appear in `A` record and `ANY` queries. For example, the `RRSIG` record is often large and these records dominate the packets in which they occur, but they are only present in just under half of the `A` records in the top 1 million responses. Attackers may consider which record types have the largest payload for the response and compose queries to elicit these responses.

Interestingly, the use of DNSSEC (`RRSIG`, `DNSKEY`, and `NSEC3` in the table) to ensure the authenticity of DNS records has the unintended consequence of adding a relatively large number of bytes to DNS responses, thereby improving DNS amplification attacks. Figure 10 shows the correlation between amplification factor and DNSSEC related bytes for the top million most amplifying responses to `A` queries in the most recent trial. Figure 11 shows the same for

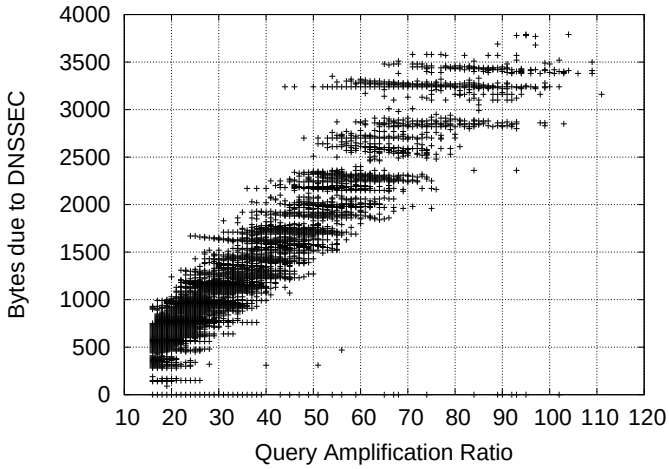

Figure 10: Scatterplot showing the relationship between query amplification ratio and the number of DNSSEC bytes in the response for the top 1 million `A` query amplifiers. For better readability, we binned the results (into 10 byte samples on the Y axis and to the nearest whole number on the X axis).

`ANY` queries. Clearly, DNSSEC is causing large increases in amplification, for both `ANY` queries and `A` queries.

We found that DNSSEC can substantially increase the size of A record queries and that this can result in high amplification ratios, as show in Figure 10. The work by Van Rijswijk-Deij *et al.* [15] found that the vast majority of `A` query amplification ratios were below their acceptable maximum (which they define in terms of DNS amplification rates without EDNS support). While our results are consistent with theirs, we focus on the top 1 million most amplifying A records and find that they represent an opportunity for attackers to achieve high amplification rates.

For brevity, we have omitted the detailed results from the other trials. At a high level, the percentage of packets containing DNSSEC records have been increasing. In the top 1 million EDNS group for domain `A` record queries, the number of responses with `RRSIG` records increased from 31.88% in July 2013 to 44.52% in Jan. 2015 before reaching 48.8% in May 2015. This increase in

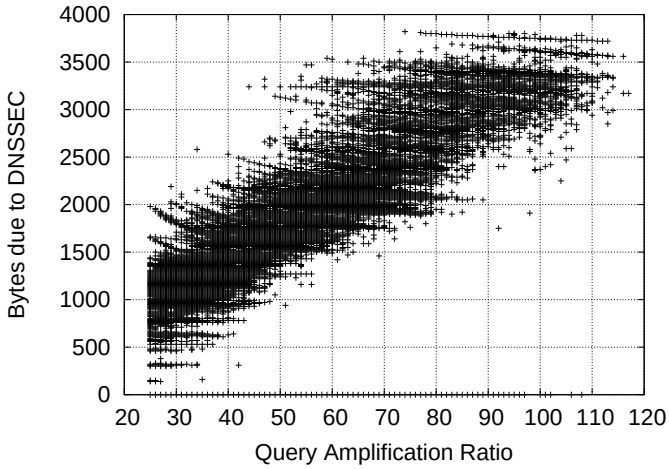

Figure 11: Scatterplot showing the relationship between query amplification ratio and the number of DNSSEC bytes in the response for the top 1 million `ANY` query amplifiers. For better readability, we binned the results (into 10 byte samples on the Y axis and to the nearest whole number on the X axis).

DNSSEC deployment appears to be driving the growth in DNS amplification potential.

While DNSSEC leads to larger responses, in particular due to the resource record signatures that are inherent to the protocol, DNSSEC provides valuable authenticity and integrity guarantees to clients. While it is important to recognize the role of DNSSEC in amplification attacks, authoritative DNS servers should continue to deploy DNSSEC due to the benefits of the protocol. However, during a DNS amplification attack, in-network security devices may choose to strip DNSSEC records to avoid saturating the destination network's bottleneck link.

4. Measuring the Adoption of DNS Rate Limiting

A recent document proposed the rate-limiting of DNS responses at the DNS server to mitigate the use of DNS amplification in practice [5]. US CERT recommended organizations employ such rate-limiting, where possible, with a

limit of five identical responses to the same origin per second [4]. However, CERT acknowledged that some popular DNS servers, notably the versions of Microsoft's DNS server before Windows Server 2016, lack response rate limiting functionality, making rate-limiting impractical for many organizations. At the time of writing, this repeated response rate-limiting is the only standardized scheme available at DNS servers. We thus focus our measurement study on this approach.

CERT also acknowledged that rate-limiting may cause legitimate DNS queries to go unanswered if there is significant packet loss or other patterns. In our own prior work [21], where we monitored the DNS queries being issued to the authoritative servers at the Oak Ridge National Laboratory, we found that over 26,000 DNS resolvers re-issued a repeated query before the expiration of the five-minute TTL associated with the record. We found about 35% of the repeated queries were issued within the first 10 seconds of the original resolution request, likely due to DNS packet loss or fragmentation [22]. Further, we saw that some large Internet service providers load balanced their clients' DNS requests across caching DNS resolvers on contiguous IP addresses. Because the DNS rate limiting standard recommends rate-limiting at the /24 network prefix, it is possible that the combination of packet loss, load balancing, and resolvers that do not cache results will cause legitimate resolvers to exceed the rate-limit. This will deny clients access to the organization's services. Organizations have an incentive to avoid rate limiting or to set a high rate-limit value to avoid losing business or negatively affecting their customers.

To determine the impact of rate limiting, we randomly sampled 178,312,669 unique combinations of domains and authoritative name servers contained in the zone files on September 30, 2015. These represented 44% of the total domain/NS pairs in the zone files. Between October 2, 2015 and October 16, 2015, for each sampled pair, we issued a burst of 30 identical DNS queries for the domain to the name server. We recorded all of the response to files using packet capture software. We then analyzed these packet captures for signs of rate limiting.

We first looked for explicit indications that rate limiting was being employed.

If any trial produced a reply containing any new error or a truncation flag, we considered it as evidence that the query triggered a rate limiting response and recorded the rate limit as being triggered. We then looked for indications of repeated queries being silently dropped. We excluded any queries in which all trials failed, since that is more likely a sign of misconfiguration or server failures than rate limiting. For the remaining queries, we examined each trial to determine if there was a pattern of dropped replies or if the reply packet had an error or truncation flag set. If the query trials resulted in five or more of the 30 expected reply packets being missing, we considered that as evidence of rate limiting and counted the rate limit as being the occurrence in which the first packet was lost.

In Figure 12, we show the percentage of queries that showed rate limiting and what packet number first triggered the rate limiting response. For the fraction of domains that do employ rate limiting, there are jumps at more than 1 and more than 5. No other values appeared to be shared by significant percentages of the rate limiters. While the CERT suggested limit of five repeated queries may have influenced a portion of the deployers, there does not appear to be a consensus amongst deployers for the appropriate value.

We now examine rate limiting impact assuming a threshold of 25 repeated queries, since there is no clear cutoff for the number of queries triggering rate limiting, and 25 is the highest number that would trigger our five dropped packets rule. We see that 18,241,886 of the queries reached servers that employ a rate limit of 25 repeated queries or less. That figure represents 10.23% of the 178,312,669 domain-server pairs examined, implying that attackers could rapidly repeat almost 90% of desired attack queries up to 25 times per second without triggering any defensive response from the authoritative DNS server.

These results suggest that rate-limiting is infrequently used in practice. As a result, it is unlikely to be a significant factor in mitigating DNS amplification attacks.

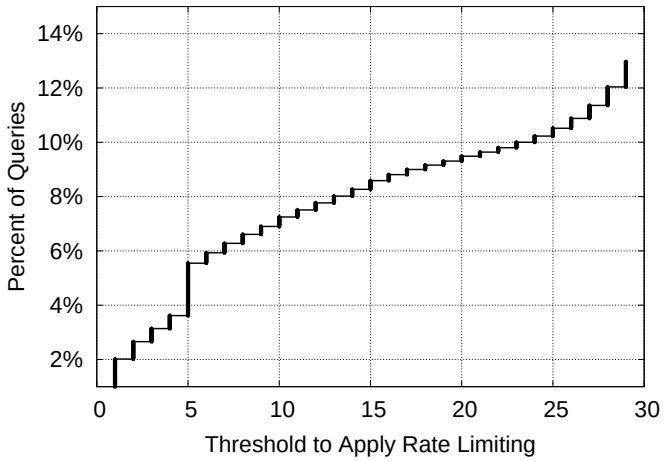

Figure 12: This CDF shows the number of unique domain to name server pairs that employed rate limiting after the indicated number of queries.

5. Discussion

We now discuss the impact of open DNS resolvers and provide recommendations for organizations.

5.1. Impact of Open DNS Caching Resolvers

For an attacker, an open DNS resolver has the functional impact of being another system that can be used to issue high amplification queries. In particular, the attacker can use the open resolver to issue a query to an authoritative server, knowing that the response will be of a known, high amplification factor. In particular, the attacker can create a server with specifically crafted records that will yield high amplifications. When asked about the query in the future, the resolver will return the result from cache, replicating the high amplification. This allows an attacker to bypass any rate limiting employed at the authoritative server, since the resolver serves the result from cache. Accordingly, open resolvers allow attackers to use a high volume query repeatedly and distribute the attack source to additional machines.

While open resolvers may be convenient for an attacker, allowing load balancing and a greater number of highly amplifying reflectors, they are ultimately unnecessary. In practice, few authoritative servers employ any form of rate limiting and tens of thousands of authoritative servers offering average amplification factors exceeding 14x for even the most common query types.

Since open DNS resolvers can be used to distribute an attack, we recommend that operators close their DNS resolvers where possible. This change is often straightforward in DNS server software and restricts the IP address range that may use the DNS resolver. However, some DNS administrators may be unaware of the implications of open DNS resolvers and may use default settings or remove such restrictions to expediently bring the resolver online. Accordingly, we encourage the community to also consider efforts that may yield greater gains. In particular, we encourage community members to identify the DNS queries that yield the greatest amplification and focus on strategies to curtail such amplification. System administrators can evaluate their own DNS zones by allowing a test machine to issue an `ANY` query for each host name in their own zone files and evaluate the response sizes to identify queries that yield high amplification factors. External entities can perform queries similar to our own to identify highly amplifying queries.

By reducing the maximum amplification factor of any DNS query from the current 41x to a lower amount, such as 20x or less, the DNS amplification potential of all open DNS resolvers would be halved without having to close a single open resolver. Essentially, attackers would be unable to query and cache such highly amplifying records at open resolvers and their attack potential would decrease. Importantly, our distributions show that alterations to a relatively small number of highly amplifying authoritative servers could yield significant benefits.

5.2. Recommendations

As shown in Section 3.3, the greatest amplifications were observed when using `ANY` queries. This query type resulted in amplifications approximately

double those observed with `A` record queries. There are few if any legitimate uses for the `ANY` query type. We recommend DNS software providers disable `ANY` records by default and we recommend that deploying organizations filter responses to `ANY` queries, both in authoritative and recursive responses, unless necessary for a particular environment. Based on the difference in response rates we received for `A` and `ANY` queries, it seems that some servers are already refusing to answer `ANY` queries. Further, organizations can proactively filter any DNS responses to the `ANY` DNS query type as a defensive mechanism [23]. Filtering the `ANY` response is unlikely to cause problems for legitimate applications, but such a feature is not present in some widely-used DNS server implementations and would require a manual implementation, via a firewall or similar mechanism.

We also recommend that organizations consider the DNS records they expose publicly. In prior studies [24], [18], we found that organizations may configure their zones incorrectly, exposing information that may be intended only for internal use. Some DNS server software supports access control lists, which can be used to only provide zone information to some IP addresses. By limiting the information exposed on a server, it becomes less likely that an attacker will use that server as part of an amplification attack.

Ultimately, attackers will still be able to use DNS reflection and amplification attacks until IP address spoofing is completely eliminated. However, by closing open DNS resolvers, eliminating ANY records, employing rate limiting, and carefully examining DNS responses to reduce their size, defenders can reduce the amount of amplification associated with DNS responses and thus lessen the impact of these attacks.

6. Conclusion

In this work, we analyze the attack potential associated with DNS amplification attacks that focus on using authoritative servers as amplifiers. We find that attackers can launch damaging attacks with relatively little bandwidth requirements at the attack traffic source. We further find that attackers could scale

up such attacks easily. We find that less than 3.8% of authoritative servers are responsible for the highest amplification factors in some data sets. Further, we note that DNSSEC played a significant role in amplification: by securing the DNS infrastructure, defenders are increasing the amplification potential of DNS reflector attacks. Further, we note that DNS response rate limiting has roughly 10% adoption. Accordingly, by repeatedly querying each DNS server that does not employ rate limiting, attackers could easily launch massive floods using only authoritative servers.

While much discussion has focused on open resolvers, they functionally serve as distributed mirrors of the top amplifying authoritative servers. These resolvers could also let attackers bypass rate-limiting at servers; however, with relatively few servers using rate-limiting, open resolvers only seem valuable to have a larger base to distribute attacks.

We note that organizations may be able to decrease their role in DNS amplification attacks by rate-limiting DNSSEC responses when repeatedly queried by a single source or by disabling certain DNS query types, such as the `ANY` record.

[1] L. Constantin, "Attackers use DNSSEC amplification to launch multi-vector DDoS attacks," http://www.computerworld.com/article/3097364/security/attackers-use-dnssec-amplification-to-launch-multi-vector-ddos-attacks.html, July 2016.

[2] B. Brenner, "DNSSEC targeted in DNS reflection, amplification DDoS attacks," Akami SIRT Alert, February 2016.

[3] F. Ricciato, A. Coluccia, and A. D'Alconzo, "A review of DoS attack models for 3G cellular networks from a system-design perspective," *Computer Communications*, vol. 33, no. 5, pp. 551–558, March 2010.

[4] US-CERT, "DNS amplification attacks," Alert (TA13-088A): https://www.us-cert.gov/ncas/alerts/TA13-088A, July 2013.

[5] P. Vixie and V. Schryver, "DNS response rate limiting (DNS RRL)," http://ss.vix.su/~vixie/isc-tn-2012-1.txt, April 2012.

[6] Center for Measurement and Analysis of Network Data, Naval Postgraduate School, "Spoofer project: State of IP spoofing," http://spoofer.cmand.org/summary.php, February 2014.

[7] US-CERT, "Smurf IP denial-of-service attacks," Advisory (CA-1998-01): http://www.cert.org/historical/advisories/CA-1998-01.cfm, January 1998.

[8] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 3, pp. 38–47, 2001.

[9] US-CERT, "NTP amplification attacks using CVE-2013-5211," Alert (TA14-013A), January 2014.

[10] M. Prince, "Technical details behind a 400gbps NTP amplification DDoS attack," http://blog.cloudflare.com/technical-details-behind-a-400gbps-ntp-amplification-ddos-attack, February 2014.

[11] C. Rossow, "Amplification hell: Revisiting network protocols for DDoS abuse," in *Network and Distributed System Security (NDSS) Symposium*, 2014.

[12] M. Kührer, T. Hupperich, C. Rossow, and T. Holz, "Exit from hell? reducing the impact of amplification DDoS attacks," in *USENIX Security Symposium*, 2014.

[13] L. Krämer, J. Krupp, D. Makita, T. Nishizoe, T. Koide, K. Yoshioka, and C. Rossow, "Amppot: Monitoring and defending against amplification ddos attacks," in *Research in Attacks, Intrusions, and Defenses*. Springer, 2015, pp. 615–636.

[14] D. C. MacFarland, C. A. Shue, and A. J. Kalafut, "Characterizing optimal DNS amplification attacks and effective mitigation," in *Passive and Active Measurement Conference*, 2015.

[15] R. van Rijswijk-Deij, A. Serotto, and A. Pras, "DNSSEC and its potential for DDoS attacks," in *ACM Internet Measurement Conference*, 2014.

[16] J. Damas and F. Neves, "Preventing use of recursive nameservers in reflector attacks," IETF RFC 5358, October 2008.

[17] R. Elz, R. Bush, S. Bradner, and M. Patton, "Selection and operation of secondary DNS servers," IETF RFC 2182, July 1997.

[18] A. J. Kalafut, C. A. Shue, and M. Gupta, "Touring DNS open houses for trends and configurations," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, p. 1, 2011.

[19] J. Damas and P. Vixie, "Extension mechanisms for DNS (EDNS(0))," IETF RFC 6891, April 2013.

[20] D. Conrad, "Indicating resolver support of DNSSEC," IETF RFC 3225, December 2001.

[21] C. Shue and A. Kalafut, "Resolvers revealed: Characterizing DNS resolvers and their clients," *ACM Transactions on Internet Technology (TOIT)*, vol. 12, no. 4, July 2013.

[22] N. Weaver, C. Kreibich, B. Nechaev, and V. Paxson, "Implications of netalyzr's DNS measurements," in *Workshop on Securing and Trusting Internet Names (SATIN)*, 2011.

[23] O. Gudmundsson and M. Majkowski, "Standard way for authoratitive DNS servers to refuse ANY query," IETF Internet Draft draft-ogud-dnsop-any-notimp-00, March 2015.

[24] A. Kalafut, C. Shue, and M. Gupta, "Understanding implications of DNS zone provisioning," in *ACM Internet Measurement Conference*. ACM, 2008, pp. 211–216.