

On the Potential of IPv6 Open Resolvers for DDoS Attacks

Luuk Hendriks¹(✉), Ricardo de Oliveira Schmidt¹, Roland van Rijswijk-Deij²,
and Aiko Pras¹

¹ Faculty of Electrical Engineering, Mathematics and Computer Science,
University of Twente, Enschede, The Netherlands
{luuk.hendriks,r.schmidt,a.pras}@utwente.nl
² SURFnet BV, Utrecht, The Netherlands
roland.vanrijswijk@surfnet.nl

Abstract. Distributed Denial of Service (DDoS) attacks have become a daily problem in today's Internet. These attacks aim at overwhelming online services or network infrastructure. Some DDoS attacks explore open services to perform reflected and amplified attacks; and the DNS is one of the most (mis)used systems by attackers.

This problem can be further aggravated in the near future by the increasing number of IPv6-enabled services in the Internet. Given that the deployment of IPv6-enabled services is increasing, it becomes important to find vulnerable IPv6 open services that could be (mis)used by attackers, and prevent that misuse. However, unlike with IPv4, simply scanning the IPv6 address space to find these open services is impractical.

In this paper we present an active measurement approach to enumerate a relevant list of open resolvers on IPv6 in the wild that could be potentially exploited in a DDoS attack. Based on the assumption that IPv6 open resolvers can be found via IPv4 ones, we show that IPv6-based amplified DDoS attacks are a significantly potential threat in the Internet: the analyzed resolvers, of which 72% are assumingly infrastructural servers, showed a median amplification factor of 50.

1 Introduction

One of the most prevalent and noticeable types of attacks in our Internet today is the Distributed Denial of Service (DDoS) attack. Based on reports from Akamai [2] and Arbor Networks [3], we see an increase in both number and size of these attacks. The attacks come in many forms, with the DNS-based variant being one of the most observed. This type of attack is possible because of DNS open resolvers in the Internet, which accept DNS queries from any source. By spoofing the source IP of a DNS request with the target's address, an attacker is able to deceive an open resolver, which ultimately answers directly to the target, constituting a reflected DDoS (DRDoS) attack. Furthermore, as the DNS response can be many times larger than the request, there is a form of amplification in the attack. These phenomena combined result in a type of threat that

is effective and hard to mitigate, with direct consequences for both operators and end-users, *e.g.* significant decrease in quality of experience. It is therefore important for operators to be aware of open resolvers in their own networks, to fix them and prevent others from mis-using them in such an attack. Finding these open resolvers on IPv4 is feasible, and has been subject of multiple studies [13, 14]. Tools and services [1, 7] to find these open resolvers have existed for years. As these approaches rely on scanning the entire address space, they are not applicable in the IPv6 Internet. With this work, we present an approach to find open resolvers with IPv6 connectivity, and analyze their potential for attacks.

We assume that a certain share of open resolvers on IPv4 have a form of IPv6 connectivity, and are also resolving openly over IPv6. Besides dual-stacked hosts, running resolver software responding on both protocol versions, we expect to find *infrastructural DNS resolvers*: machines deployed by network operators to handle DNS resolution for their customers, but which are not directly used by the customers. Instead, a forwarding resolver in front of the actual resolving infrastructure is taking DNS questions and sends the answers to these customers, while the infrastructural resolvers perform the actual resolving. This infrastructural part should not be accessible for customers inside the network, let alone from connections outside of that network. Our hypothesis is that operators forget to ACL/firewall the IPv6 part of their resolving infrastructure, effectively enabling misuse. As tooling and services to find open resolvers lack support to find resolvers with IPv6-connectivity, most operators will be unaware of open resolvers in their networks. In order to find open resolvers with IPv6 connectivity, we present an active measurement approach (Sect. 3) based on querying a zone where the authoritative nameserver is only reachable over IPv6. With the results from that, we conduct additional experiments to analyze whether these are indeed infrastructural DNS resolvers.

Contributions: We present a novel methodology to find open resolvers on IPv6, and validate it by performing measurements using the complete IPv4 address space. Consequently, we show that finding open resolvers on IPv6 using our approach is feasible. Our analysis shows roughly 70% of the found resolvers are infrastructural, thus likely to have good connectivity and high bandwidth. Furthermore, we show that queries generate large answers over the found IPv6 paths, with amplification factors of over 100 for the top 5%. These findings emphasize the need for awareness, wherefore we will approach anti-abuse projects to share our code with for adoption. We believe incorporating the code in well-known, existing efforts will have the most effective impact. For ethical reasons, we do not publish our code: it will be shared with fellow researchers and interested anti-abuse projects on a request basis.

First, we will sketch out (Sect. 2) possible DNS resolver setups, and explain why our approach can determine their possible IPv6 connectivity. Our methodology (Sect. 3) describes the measurement setup (Sect. 3.2), the steps to obtain open resolvers on IPv6 (Sect. 3.3), measurements to identify infrastructural resolvers (Sect. 3.4), and an analysis of possible amplification (Sect. 3.5).

Then, we discuss (Sect. 5) our approach and findings, and list related work (Sect. 6). Lastly, we conclude (Sect. 7) that open resolvers on IPv6 have, although relatively low in number, a large potential for severe DDoS attacks.

2 Background

2.1 Using DNS to Traverse from IPv4 to IPv6

The approach in this work is based on normal behavior of the Domain Name System (DNS), in terms of resolving hostnames: a client sends a query to a resolver, which collects the required information at one or more authoritative nameservers. The resolver constructs the answer and sends it back to the client. The only trick is a special configuration of certain nameservers, making them only reachable over either IPv4 or IPv6, but not both. It is important to emphasize that we are dealing with two different forms of ‘IPv4’ and ‘IPv6’: the process involves Resource Record (RRs) for both, *i.e.* A and AAAA records, but we are interested in the protocol that is actually used for transport.

Using `example.v6only.ourdomain.net` as an example, where the `v6only` zone is delegated to a nameserver only reachable over IPv6, the following steps take place in the resolving process:

1. The client asks the resolver, over IPv4, for the A record of the domain.
2. The resolver contacts the `.` (root) and `net.` server, to find out where the authoritative nameserver of `ourdomain.net` is.
3. The resolver contacts the nameserver of `ourdomain.net`, asking for the NS record of the `v6only.` subdomain, in order to find out who to ask for anything under that subdomain. The NS record contains `ns6.ourdomain.net`, for which only an AAAA record exists.
4. The resolver tries to contact that nameserver on the IPv6 address from the AAAA record: only in case the resolver has IPv6 connectivity, traffic arrives at the nameserver.

Thus, while initially contacting the resolver over IPv4, eventually packets over IPv6 will arrive on the authoritative side—if and only if the resolver has any form of IPv6 connectivity. This way, **by using DNS on the application layer, we traverse from IPv4 to IPv6 on the network layer.**

2.2 Possible Resolving Setups

In practice, the aforementioned resolver is not necessarily a single entity. Multiple machines can form a resolving infrastructure, including *e.g.* load-balancers, without any ostensible difference for the end-user.

In our search for resolvers with forms of IPv6 connectivity, we generalize and consider two scenarios, as depicted in Fig. 1. The simple form Fig. 1a features a single host for the resolving, which is thus dual-stacked and both IPv4 and IPv6 connections are instantiated by that host itself. Examples of this scenario

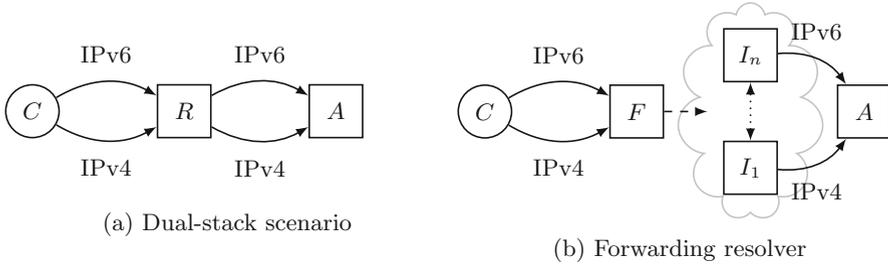


Fig. 1. Generalized scenarios of DNS resolving setups. *C*: client, *R*: resolver, *F*: forwarding resolver, I_i : infra, *A*: auth. nameserver.

are (badly configured) Customer Premises Equipment (CPE) handling queries on their WAN-side, or a Virtual Private Server (VPS) running resolver software. In case of Fig. 1b, the resolver used by clients is not performing full resolving itself, but rather forwards queries to one or more upstream resolvers. In this case, IPv4 and IPv6 connections towards authoritative nameservers are not necessarily originating from one and the same machine.

3 Methodology

3.1 Finding IPv4 Open Resolvers

The first step in our approach is to enumerate open resolvers on IPv4 available in the Internet, which will later be tested for IPv6-connectivity (Sect. 3.2).

To find open resolvers in the Internet, we scan the routable IPv4 address space. In this scan we simply send out DNS queries to every IPv4 address and wait for incoming responses. However, the fact that a response is received does not necessarily mean that the replying open resolver can be somehow misused; that is, we distinguish responses where DNS resolution is not explicitly refused. To do so, we look into the returned RCODE¹, where RCODE 5 when the server refuses to answer: those are filtered out and not further acted upon. To maximize our results, we are liberal with other RCODEs. Our scans are based on `zmap` [7] and its DNS module, with an adaption to accept responses from unexpected ports, again to maximize results.

As we expect to find *e.g.* CPEs subject to time (DHCP-leases, IPv6 address lifetimes), we perform our measurements directly after finding an open resolver on IPv4: this, combined with the aforementioned liberal selection criteria, makes existing available lists of open resolvers unfit for our research. We go into more ethical considerations of our measurements in Sect. 5.1.

¹ <http://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-6>.

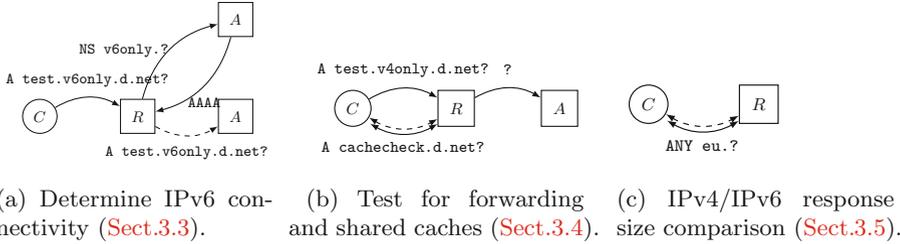


Fig. 2. Visualization of methodology steps, per phase. *C*: client, *R*: resolver (abstracted, see Fig. 1), *A*: authoritative. Solid lines depict IPv4 transport; dashed lines IPv6 transport.

Table 1. Configuration of DNS zone for all measurements.

v6only.ourdomain.net	NS	ns6.ourdomain.net.
ns6.ourdomain.net	AAAA	2001:db8::53
dns6ver.ourdomain.net	AAAA	2001:db8::53
v4only.ourdomain.net	NS	ns4.ourdomain.net.
ns4.ourdomain.net	A	123.123.123.123
dns4ver.ourdomain.net	A	123.123.123.123
cachecheck.ourdomain.net	A	123.123.123.123

3.2 Measurement Setup

With a list of open resolvers on IPv4 at hand, we start the actual measurements, divided in three steps (Fig. 2). In the following subsections, we detail the three phases, which all involve specifically configured resource records in the DNS. An overview of this configuration is given in Table 1. It is important to understand that there are two different uses of IPv4 and IPv6 in our approach. The DNS protocol specifies Resource Records of type A and AAAA designated to IPv4 and IPv6, respectively. Our interest is, however, in the IP protocol-version used for transport.

3.3 Determining IPv6 Connectivity

First, we determine whether the open resolver (found in Sect. 3.1) has any form of IPv6 connectivity (Fig. 2a). Every open resolver is queried over IPv4 for a specific qname under a zone for which the nameserver has only an AAAA-record, thus no A-record: `$ipv4.$timestamp.v6only.ourdomain.net`. If we observe the query arriving at the authoritative side, we can extract the initially queried resolver from the qname (*i.e.* `$ipv4`), and we know it has some form of IPv6-connectivity. To verify whether the IPv6 address we have thus uncovered is itself an open resolver, we send it a verification query: `$ipv4.$timestamp.dns6ver.ourdomain.net`. The initial `$ipv4` is still

included for ease of analysis. Once that query is observed at the authoritative side, we know we found an open resolver, and we continue with the next two steps. The `$timestamp` is used to distinguish different runs of measurements, and to prevent any forms of caching.

3.4 Distinguishing Dual-Stack and Infrastructural Setups

Now that we have pairs of IPv4 and IPv6 addresses belonging to a resolving entity, we perform additional queries (Fig. 2b) to gain insight in how this resolving entity is set up, distinguishing the two scenarios depicted in Fig. 1. Firstly, the IPv4 address is queried again, but now for a zone that is only reachable over IPv4: `$ipv4.$timestamp.v4only.ourdomain.net`. Upon the query incoming at the authoritative side, comparing the connecting IPv4 address and the initially queried address (*i.e.* `$ipv4`) tells us whether we are dealing with a single machine, or whether forwarding or distribution has occurred. Secondly, we test for a shared cache between the IPv4 and IPv6 addresses. For each pair of IPv4 and IPv6 addresses, both addresses are queried for the same qname, based on a hash of both addresses and the measurement timestamp: `h($ipv4$ipv6$timestamp).cachecheck.ourdomain.net`. This query is performed twice over IPv4, and twice over IPv6. All the four queries are 5s apart. Based on the TTL values in the answers, we can determine whether the resolver is actually caching on any or both of the protocols, and whether that cache is shared.

3.5 Comparison of Response Sizes for IPv4 and IPv6

Finally, as shown in Fig. 2c, the response sizes of pairs of IPv4 and IPv6 addresses are compared. We aim at large responses, so queries are DNSSEC-enabled and ask for the ANY-record [16]. We do not use TCP fallback. Queries of this form for `com.` and `eu.` are sent to both the addresses. We capture the incoming packets with their full payload in order to find explanations for differences in response sizes.

Processing on the Authoritative Side. If a queried IPv4 open resolver has a form of IPv6 connectivity (or delegates the resolving to a host that has IPv6 capabilities), the constructed query ends up on the host with the IPv6-address configured in the AAAA record. From incoming queries, we extract the information listed in Table 2.

4 Results

4.1 What Share of the Resolvers Generate IPv6 Traffic?

Our measurement yielded 1038 *unique* IPv6 addresses, verified to be openly resolving. This number is distilled from 78698 unique pairs of IPv4 and IPv6

Table 2. Information extracted from incoming queries.

v6	IPv6 source address of query that reached our nameserver
qname	Queried name
qtype	Query type (should be A)
orig_ts	Timestamp extracted from qname
orig_v4	IPv4 address of the server initially queried, extracted from qname
asn4	ASN of orig_v4
asn6	ASN of v6

Table 3. Overview of measurement results

IPv6 connectivity	1.49M unique pairs		
Open on both IPv6/IPv4	78698 (5.3%) unique pairs		
of which unique IPv6	1038 addresses	745	(72%) infrastructural
		922	(89%) caching
of which unique IPv4	72784 addresses	258	(0.4%) infrastructural
		7486	(10%) caching
		55582	(76%) mismatches

addresses—of which both IPv4 and IPv6 addresses were openly resolving. In these pairs were 72784 unique IPv4 addresses, of which (based on the queries for the v4only zones) 76% did not match with the address contacting our authoritative nameserver. Upon verifying whether these *mismatches* were openly resolving, we found 258 IPv4 addresses to do so. These are what we call *infrastructural resolvers* (Sect. 2.2): 745 (72%) of the 1038 IPv6 addresses were associated with these. An overview of these numbers is given in Table 3.

In total, we found more than 1.49M unique pairs of IPv4 and IPv6 addresses to *generate* a form of IPv6 traffic, *i.e.* we observed incoming packets from the IPv6 address after sending a query to the IPv4 address. The verification query (dns6ver) reduced this to the aforementioned 78698 address-pairs (5.3%).

4.2 Caching Characteristics

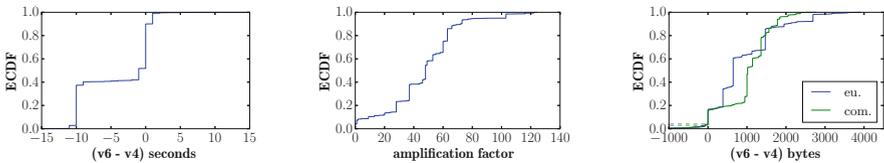
Comparing the Time-to-live (TTL) values in answers for the cachecheck queries showed that for the 1038 unique IPv6 resolvers, 922 (89%) did cache answers. For pairs of IPv4 and IPv6 addresses that both cache, nearly 60% do not share their cache, as can be seen in Fig. 3a: for each pair, the TTL of the cachecheck answer over IPv4 is subtracted from the TTL of the answer over IPv6. As these queries were sent 10 s apart, the peak at -10 in the plot implicates 40% shared caches. The long tail can be explained by resolvers overwriting the actual TTL with their own minimal values, *e.g.* 600 s, on IPv4, while the IPv6 resolver respected the value configured in our zone, *i.e.* 60 s. (Note that the

40% is a conservative number as infrastructures can comprise multiple upstream resolvers, thus requiring multiple queries to detect shared caches.)

4.3 Amplification Factor

Looking into the achievable amplification factor, we measured the response sizes of the answers to our ANY queries. The distribution, Fig. 3b, shows the responses over IPv6 feature significant amplification. The median amplification factor is 50, whereas the top 5% is amplified more than 100 times.

Comparing the response sizes over IPv6 with those over IPv4 requires consideration, as the found IPv6 addresses belong to machines that are often different from the initial IPv4 open resolvers. While this does not allow us to draw general conclusions on the network layer protocols, it does provide insight on how much one with malicious intents can gain (in terms of amplification) when the transition from IPv4 to the IPv6 resolver is made. We compared the response sizes to the ANY queries for each *pair* of IPv4/IPv6 addresses, and show the difference in bytes in Fig. 3c. The dashed horizontal lines emphasize where the difference in response size is exactly 0 bytes, *i.e.* the response sizes are equal over both IPv4 and IPv6. The figure shows that, for the analyzed pairs, 90% of the answers over IPv6 are equal or bigger in size than the answers coming from the IPv4 address.



(a) TTL difference for servers caching on both IPv4 and IPv6.

(b) Distribution of amplification factor over IPv6.

(c) Difference between response sizes for pairs of v4/v6 resolvers for ANY.

Fig. 3. Analysis of caching, amplification and response size characteristics.

4.4 Distribution of Open Resolvers per Network

We looked further into which networks² the open resolvers reside in. Counting the number of unique IPv6 addresses acting as open resolvers per Autonomous system (AS), we find the top 10 to account for more than half of all the IPv6 open resolvers: the other half is spread over 216 different networks. Figure 4 lists these top 10 networks, showing their share of the total number of found open resolvers. The AS with most unique resolvers (accounting for almost 9% of the total) is a South-Korea based Internet service provider (ISP). Number 2 is the backbone of a mobile operator in Germany. The top 3 is completed by

² IP to ASN resolving done using *pyasn* with CAIDA RouteViews data. Network names and country codes obtained from Team Cymru.

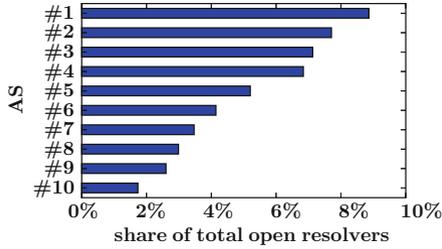


Fig. 4. Top networks with most unique open resolvers (IPv6). The 10 networks in this graph account for 51% of all open resolvers.

a French hosting company. The remainder of the top 10 consists of a mix of service providers and hosting companies, with the notable exceptions 6 and 7: there we find a public DNS resolver service from the US, and an organization famous for providing IPv6 tunnel solutions, also from the US. Geographically, there is not a definitive domination by any continent, although without 6 and 7, we are mainly left with networks from Western Europe and Asia. Aggregation on country indeed shows mainly countries from those continents (Table 4).

Table 4. Top 10 countries with most unique open resolvers (IPv6), accounting for 78% of all.

Country	Unique	% of total
Germany	186	17.9%
United States	150	14.5%
South Korea	104	10.0%
France	99	9.5%
Taiwan	78	7.5%
Mexico	72	6.9%
China	53	5.1%
Thailand	25	2.4%
Hong Kong	22	2.1%
Sweden	22	2.1%

4.5 Interface Identifier Analysis

From all unique IPv6 addresses found to be openly resolving, more than half are assumed to be configured by a human, strengthening the likeliness of these being infrastructural resolvers. For this, we look at the Interface Identifier (IID), the last 64 bits of the IPv6 address. Out of the 1038 addresses, 622 had non-zero bits only in their last hextet: all other of the 64 last bits of the address were 0. Of those, 570 (*i.e.* 55% of all) feature only decimal characters—no hexadecimals—in

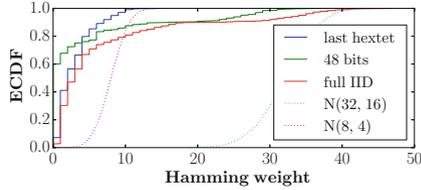


Fig. 5. Hamming weight distribution of (parts of the) IIDs (Color figure online)

Table 5. Characteristics of IID of resolver addresses

Total IPv6 addresses	1038	
Hex in IID	225	
of which SLAAC (ff:fe)		83
All 0 but last hextet	622	
of which decimal hextet		570

their “natural” notation. The addresses with hexadecimals were for 37% identifiable as Stateless Address Auto-Configuration (SLAAC) addresses. These numbers are further detailed in Table 5. The distribution of the hamming weights of the IIDs is far below a normal distribution. As shown in Fig. 5, where based on the central limit theorem a mean of 32 is to be expected for the last 64 bits (blue dotted line), we find only 10% of the addresses (solid red line) to feature that hamming weight. The solid blue and green lines depict the distribution of respectively the last hextet, and the 48 bits before that last hextet. For reference, the normal distribution of the last hextet is also plotted (pink dotted line).

5 Discussion

5.1 Ethical Considerations

Presenting a methodology that can be misused for malicious intents might raise ethical concerns. It is however comprised of technologies and configurations that are not new themselves, and have been available to anyone for a long time. Furthermore, we emphasize that the traversal from IPv4 to IPv6 using DNS will only reveal IPv6-connected resolvers, but does not enable direct use over IPv4, significantly reducing the opportunity to misuse found hosts.

Furthermore, in our measurements, we queried for the domain of the university (*i.e.* utwente.nl), to hint at the benign intent of our doings.

5.2 Pitfalls in Scanning/Great Firewall of China

Using zmap, the default query is A for google.com. Using this yielded $\sim 220\text{M}$ “open resolvers”, which is not in line with literature. Using A, utwente.nl yields a

far lower number. Initial analysis of this difference points us to a large share of IP addresses from Chinese networks. Those IP addresses acted as open resolvers in the sense that they seemingly returned answers on our DNS questions. However, they only do so for specific qnames, like google.com, while for utwente.nl no response was sent. Furthermore, when querying a subset of these IP addresses by hand, we observed responses to be incorrect—random to a certain degree. When querying for AAAA records, the responses contain invalid IPv6 addresses or (again, random) IPv4 addresses. Based on the large number of these fake resolvers, and their location, we reckon to have hit a network-level, government-managed entity.

5.3 Response Size Difference

The difference in response sizes has multiple explanations. Analysis of the full packet capture of the answers on our ANY queries shows $1.3\times$ more answers over IPv6 than over IPv4. Of the answers over IPv4, 60% is *malformed*: more than 99% of these malformed answers are exactly 512 bytes in size, hinting at truncation of packets, possibly by middleboxes. On the contrary, no malformed answers were observed over IPv6. Looking at valid answers, we find 71% to be empty (*i.e.* ANCOUNT 0) over IPv4, versus 6% over IPv6; this likely indicates different configuration on the application level.

6 Related Work

To the best of our knowledge, we are the first to systematically investigate the potential of IPv6 open resolvers in the context of DDoS. However, complementary to our work, there are many studies that addressed the DDoS problem in multiple ways. In 2014, Welzel *et al.* [17] found more than 60% of targets of botnet-driven DDoS attacks to be impacted significantly. More recently, Moura *et al.* [9] assessed the impact of DDoS attacks against the Root DNS in Nov. and Dec. 2015, showing how the distribution of the root system allowed for resilience. Other works focused on individual aspects of DDoS, such as the amplification factor. In 2014, Rossow [12] found that 14 UDP-based protocols are susceptible to bandwidth amplification with a factor up to 4670; and later in 2015, Kühner *et al.* [10] collaborated in a large scale campaign to reduce the number of vulnerable NTP servers by more than 92%. Also in 2014, Czyz *et al.* [6] showed that there were 2.2M potential NTP amplifiers in the Internet, some replying to probes with several gigabytes of data; and van Rijswijk-Deij *et al.* [16] showed that DNSSEC-signed domains can result in very high amplification factors with responses $59\times$ larger (and $179\times$ in some cases). In 2015, MacFarland *et al.* [11] addressed the potential of amplification by authoritative DNS nameservers, showing that very few nameservers are responsible for the highest amplification factors.

On another angle, many studies have also addressed IPv6 measurements. Beverly *et al.* [5] present an active approach to identify shared IPv4 and IPv6 infrastructures in the Internet. Using a controlled authoritative nameserver,

Berger *et al.* [4] studied the relation between IPv4 and IPv6 DNS resolvers. A similar approach was used by Schomp *et al.* [13] to study the behavior of DNS servers in terms of caching and handling of TTL.

Finally, concerning IPv6 scanning, Ullrich *et al.* [15] proposed an active approach on the assumption that addresses are systematically assigned; they were able to identify a large number of active IPv6 addresses, although likely far from a realistic address census. Gasser *et al.* [8] proposed a hybrid active/passive approach by creating a hitlist, at the time containing 150M unique IPv6 addresses.

7 Conclusions

In this paper, we prove finding open resolvers with IPv6 connectivity is feasible. We leverage the fact that we can scan the entire IPv4 address space, and combine that with the traversal of IPv4 to IPv6 using the higher layer DNS protocol. With this approach, we prove that one can find both dual-stacked resolvers, as well as open resolvers that are part of a resolving infrastructure.

Comparing open resolvers on the infrastructure side, we see roughly three times more IPv6 resolvers than on IPv4, suggesting improper configuration is indeed more often the case for IPv6 resolvers than for their IPv4 counterparts. And while being open on IPv6 is likely to be a form of improper configuration on the network layer (firewall/ACL), the differences in response sizes are likely also caused by configuration errors on the application layer, *i.e.* missing parameters in the resolver software specifically for IPv6. Operators do have to pay attention to multiple layers to solve this problem adequately.

From the perspective of misuse, thus comparing the found IPv6 resolvers to the far larger number of IPv4 (forwarding) resolvers, there nonetheless is reason to be concerned: one may assume infrastructural resolvers to be connected via at least 1 G, or even 10 G links. This, combined with the larger response sizes, makes for very potent attack sources. A significant share of the found resolvers cache responses, making them more effective as they do not have to query authoritative nameservers that may implement Request Rate Limiting (RRL) on their part.

By sharing our measurement code with projects that enumerate open resolvers on IPv4, we attempt to create awareness for operators, and an accessible way for them to prevent their infrastructure from being misused in attacks.

References

1. Open Resolver Project (2016). <http://openresolverproject.org>
2. State of the Internet/Security. Technical report, Akamai, Q2 (2016). <https://content.akamai.com/PG6852-q2-2016-soti-security.html>
3. WISR. Technical report, Arbor Networks (2016). <https://www.arbornetworks.com/insight-into-the-global-threat-landscape>
4. Berger, A., Weaver, N., Beverly, R., Campbell, L.: Internet nameserver IPv4 and IPv6 address relationships. In: ACM IMC (2013)
5. Beverly, R., Berger, A., Siblings, S.: Identifying shared IPv4/IPv6 infrastructure via active fingerprinting. In: PAM (2015)

6. Czyz, J., Kallitsis, M., Gharaibeh, M., Papadopoulos, C., Bailey, M., Karir, M.: The rise and decline of NTP DDoS attacks. In: ACM IMC (2014)
7. Durumeric, Z., Wustrow, E., Halderman, J.A.: ZMap: fast internet-wide scanning and its security applications. In: USENIX Security (2013)
8. Gasser, O., Scheitle, Q., Gebhard, S., Carle, G.: Scanning the IPv6 internet: towards a comprehensive hitlist. In: IFIP TMA (2016)
9. Moura, G.C.M., Schmidt, R.O., Heidemann, J., Vries, W.B., Müller, M., Wan, L., Hesselman, C.: Anycast vs. DDoS: evaluating the November 2015 root DNS event. In: ACM IMC (2016)
10. Kühner, M., Hupperich, T., Rossow, C., Holz, T.: Exit from Hell? Reducing the impact of amplification DDoS attacks. In: USENIX Security (2014)
11. MacFarland, D.C., Shue, C.A., Kalafut, A.J.: Characterizing optimal DNS amplification attacks and effective mitigation. In: Mirkovic, J., Liu, Y. (eds.) PAM 2015. LNCS, vol. 8995, pp. 15–27. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-15509-8_2](https://doi.org/10.1007/978-3-319-15509-8_2)
12. Rossow, C., Hell, A.: Revisiting network protocols for DDoS abuse. In: NDSS (2014)
13. Schomp, K., Callahan, T., Rabinovich, M., Allman, M.: On measuring the client-side DNS infrastructure. In: ACM IMC (2013)
14. Takano, Y., Ando, R., Takahashi, T., Uda, S., Inoue, T.: A measurement study of open resolvers and DNS server version. In: Internet Conference (IEICE) (2013)
15. Ullrich, J., Kieseberg, P., Krombholz, K., Weippl, E.: On reconnaissance with IPv6: a pattern-based scanning approach. In: IEEE ARES (2015)
16. van Rijswijk-Deij, R., Sperotto, A., Pras, A.: DNSSEC and its potential for DDoS attacks - a comprehensive measurement study. In: ACM IMC (2014)
17. Welzel, A., Rossow, C., Bos, H.: On measuring the impact of DDoS Botnets. In: ACM EUROSEC (2014)



<http://www.springer.com/978-3-319-54327-7>

Passive and Active Measurement
18th International Conference, PAM 2017, Sydney,
NSW, Australia, March 30-31, 2017, Proceedings
Kaafar, M.A.; Uhlig, S.; Amann, J. (Eds.)
2017, XIII, 284 p. 134 illus., Softcover
ISBN: 978-3-319-54327-7