A Large-Scale Behavioral Analysis of the Open DNS Resolvers on the Internet

Jeman Park, Rhongho Jang, *Member, IEEE*, Manar Mohaisen,
and David Mohaisen, *Senior Member, IEEE, ACM*

***Abstract*— Open DNS resolvers are resolvers that perform recursive resolution on behalf of any user. They can be exploited by adversaries because they are open to the public and require no authorization to use. Therefore, it is important to understand the state of open resolvers to gauge their potentially negative impact on the security and stability of the Internet. In this study, we conducted a comprehensive probing over the entire IPv4 address space and found that more than 3 million IP addresses of open resolvers still exist in the wild. Moreover, we found that many of them work in a way that deviates from the standard. More importantly, we found that many open resolvers answer queries with incorrect, even malicious, responses. Contrasting to results obtained in 2013, we found that while the number of open resolvers has decreased significantly, the number of resolvers providing incorrect responses is almost the same, while the number of open resolvers providing malicious responses has increased, highlighting the prevalence of their threat. Through an extended analysis, we also empirically show that the use of forwarders in the open resolver ecosystem and the possibility that incorrect or malicious responses can be manipulated by these forwarders.**

***Index Terms*— Open resolver, DNS, measurement, behavioral analysis.**

I. Introduction

THE Domain Name System (DNS) is a hierarchical distributed naming system and is a pillar of today's Internet. The primary goal of DNS is to supply a mapping between domain names and associated IP addresses. For instance, once a user types a domain name, e.g., www.example.com, into a web browser, the domain name will be mapped, by a set of DNS servers, to the associated IP address, e.g., 1.2.3.4. Almost all Internet services depend on DNS to connect users to hosts by resolving DNS queries. However, because DNS is an open system, anyone may query publicly accessible resolvers, called open resolvers. The operation of those resolvers is required in rare cases; mainly public services such as Google DNS [2] and Open DNS [3]. However, prior studies identified millions of publicly-accessible open resolvers on the Internet [4], [5]. It is shown that open resolvers are an attractive target for attackers to launch a wide variety of attacks, such as DNS amplification [6], DNS manipulation [7], *etc.*

Open resolvers can be used as a stepping stone for many attacks. For example, a report by CloudFlare highlights a 75Gbps DNS amplification DDoS attack in 2013 [8] using open resolvers in the wild. Takano *et al.* [9] also show the potential of DNS open resolvers for attacks by investigating the software version installed on those resolvers. Moreover, several previous studies demonstrated that DNS manipulation is widely used for malicious purpose by adversaries [10], [11], censorship by governments [12], or even monetary benefits [13]. These works showed that open resolvers in the wild expose their vulnerabilities to the adversaries and users alike. While DNS manipulation attacks are possible only when the open resolver is involved in the domain name resolution process, consistent with the prior work in DNS threat analysis, we pursue their enumeration and analysis (vulnerability and threat, rather than attack) for this potential threat, which could be realized in attacks eventually.

To this end, we present in this work an up-to-date view of open resolvers' threats through an in-depth analysis. Unlike the prior work that only dealt with a small subset of accessible open resolvers [7], [12], [14], [15], we attempt to investigate all open resolvers over the Internet. Moreover, we focus on the behavioral aspects of open resolvers, which provides a deeper understanding of threats posed by them. Mutual reliability is the most important factor in DNS, where a domain name is queried and a response is obtained. This reliability can be guaranteed only when a role-based behavior is performed. Observing the behavior of open resolvers is a measure of their security and DNS reliability as a whole.

A. Contribution

Our main contributions are as follows:

- We conducted a comprehensive measurement over the entire IPv4 address space to understand the behaviors and threats of open resolvers around the world by employing the prober and our own authoritative name server. An Internet-wide measurement allows us to have an empirical understanding of DNS open resolvers independent of arbitrary generalization. We found that there are about 3 million recursive resolvers that do not require any authorization for domain name resolution.
- Through quantitative analysis, we found that many open resolvers generate DNS responses in a way that deviates from the standard. More specifically, the responses from open resolvers marked fields in the DNS response header, such as the Recursion Available bit, the Authoritative Answer bit, and the response code, improperly.
- Through measurements, we report empirical results of DNS manipulation by open resolvers. By validating the open resolvers' answers, we discovered that more than

Manuscript received May 10, 2020; revised January 30, 2021 and May 28, 2021; accepted July 23, 2021; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor D. Malone. Date of publication August 26, 2021; date of current version February 17, 2022. This work was supported in part by the NSF under Grant CNS-1809000 and in part by the National Research Foundation of Korea (NRF) under Grant 2016K1A1A2912757. *(Corresponding author: David Mohaisen.)*

Jeman Park is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA.

Rhongho Jang is with the Department of Computer Science, Wayne State University, Detroit, MI 48202 USA.

Manar Mohaisen is with the Department of Computer Science, Northeastern Illinois University, Chicago, IL 60625 USA.

David Mohaisen is with the Department of Computer Science, University of Central Florida, Orlando, FL 32816 USA (e-mail: mohaisen@ucf.edu).

Digital Object Identifier 10.1109/TNET.2021.3105599

26 thousand open resolvers redirect users to malicious destinations reported as malware, phishing, *etc*.

- For a temporal contrast, we use a dataset collected in 2013. We found that the number of open resolvers has significantly decreased, while the number of resolvers manipulating responses remains the same, and the number of open resolvers providing malicious responses has rather increased. This result shows the prevalence of open resolvers as a threat, despite their decrease in number.
- We conducted a further analysis of the packets captured at the authoritative name server. By matching each flow with the same subdomain, we uncover the behaviors of open resolvers in detail, including the use of forwarder and the manipulation of answers.

B. Novelty

Our work comprehensively analyzes the open DNS resolvers and their behavior by investigating the flags and values in the header of the DNS response in-depth using a customized measurement configuration. Our work's novel aspects are not in new measurement methods but are in the methodical analyses and insights driven from this multifaceted measurement of the open resolvers and their ecosystem, using two scan snapshots over five years of time.

Although there have been multiple efforts investigating open resolvers in the wild, they failed to provide a comprehensive behavioral analysis. For example, the open resolver project (openresolverproject.org) is the first to survey open resolvers on the Internet. However, the project falls short in the following aspects. First, it does not provide any behavioral analysis of those open resolvers, and has been discontinued since 2017, reportedly due to the matureness of the space and the reduced number of open resolvers. Moreover, their final published datasets are limited to the final results returned to their prober, and not with packets captured at the authoritative name server, which would be only possible through our configuration. In this work, we show through behavioral analysis that the threat of open resolvers is persistent as evidenced by the increasing number of malicious open resolvers as of 2018, despite the overall decrease in the number of open resolvers. We also demonstrate that millions of open resolvers operate in an abnormal way (e.g., do not return the result after resolution.)

While our comparative longitudinal analysis is limited to only two snapshots, once surveyed in 2013 and the other in 2018, the insights stand on their own as a characterization, even from a historical evolution standpoint. Extending the study to other time frames remains an open question.

II. Preliminaries

This section provides a brief overview of DNS operation and the sequential process of domain name resolution.

A. DNS Resolution

The overall resolution process is shown in Fig. 1. DNS resolution begins once a user attempts to access a web service using its domain name. DNS uses caching/caches for performance, and when a domain name mapping is not cached in the local cache or the host table, the local resolver initiates a DNS query to the recursive resolver to retrieve the corresponding IP address to the domain name. The recursive resolver starts by asking root, then TLD, then the authoritative name servers.

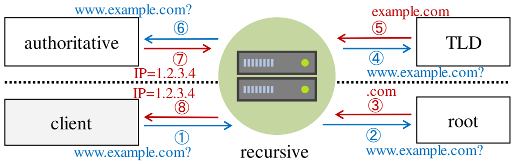


Fig. 1. DNS resolution over recursive, root, TLD, and authoritative name server. The texts and arrows in blue correspond to DNS queries, while those in red correspond to DNS answers.

Steps ② through ⑦ show the typical resolution process. The root server is the first server that receives a query from the recursive resolver, in step ②. The root servers are globally distributed and they maintain the IP addresses and location of TLD name servers. In step ③, the root name server replies to the query with the appropriate list of TLD servers for.com. In step ④ and ⑤, the recursive server sends a query for example.com and the.com TLD server responds with the IP address of the given domain's authoritative name server. In step ⑥ and ⑦, the recursive resolver communicates with the authoritative name server of example.com to find the address of www.example.com. Finally, the translated IP address of the requested domain name is forwarded to the local resolver.

B. Threat of Open Resolver

As described earlier, the recursive resolver is responsible for the recursive translation of domain names into IP addresses on behalf of clients. Among these recursive resolvers, open resolver is accessible by anyone on the Internet for resolution. Due to the role a typical recursive resolver plays in the resolution process, open resolvers are becoming a major threat to the security and resilience of the Internet. The rest of this section are details on how open resolvers are exploited; e.g., for DNS amplification attack and DNS manipulation.

1) DNS Amplification Attack: The DNS amplification attack is a DDoS attack performed by exploiting the large difference between the size of a typical DNS query and the corresponding response. Originally, DNS had a packet size limited to 512 bytes. However, due to a recent update [16], it is now possible to have more than 512 bytes in DNS responses.

'ANY' type DNS query requests information about all domains managed by an authoritative name server including 'A', 'MX', and 'CNAME'. If the authoritative name server manages a larger number of domains, the larger DNS response will be replied to the 'ANY' type query. Moreover, the standard DNS resolution is unauthenticated, which means it is possible for an adversary to generate a DNS query with a spoofed address as a source. Because the DNS response is returned to the source of the query, IP forgery would mean that someone who did not issue a given query may receive an overwhelming number of responses.

DNS amplification attacks use the above two features of open resolvers. 'ANY' type DNS queries with a victim's IP address as a source are sent to the open resolver, resulting in a concentration of DNS responses to the victim. An attacker can simply send hundreds of DNS queries to open resolvers to exhaust the victim's bandwidth without having to create a huge amount of packets for a direct DDoS attack. In a more practical way, DNS amplification attacks also can be launched by using large TXT records or querying DNSSEC signed domain. In such attacks, the open resolver acts as an attack amplifier.

2) DNS Manipulation: Another viable threat due to open resolvers is DNS manipulation. Users typically trust the results that an open resolver provides as a result of a recursive

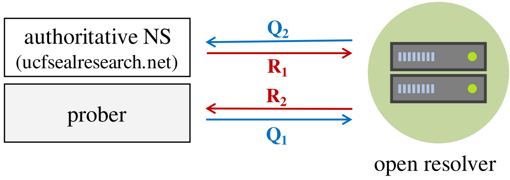


Fig. 2. DNS requests and responses among the prober, authoritative name server and open resolver. Notice that Q1 and R2 are captured at the prober by modified Zmap, while Q2 and R1 are captured at the authoritative name server by tcpdump.

resolution. In other words, the IP address contained in the DNS response is considered as a correct address of the given domain name. However, an attacker can exploit an open resolver to provide a manipulated result to the legitimate users. Instead of the genuine page the user wants to access, a false DNS response may mislead the user to a similar phishing page created by the attacker to distribute a malicious program or to steal one's credential. Even when the attacker does not own the open resolver, he may produce the same effect by injecting the manipulated record into other existing open resolvers.

III. Methodology

The goal of this work is to answer the following questions. 1) How many open resolvers exist over the world? 2) Do open resolvers behave correctly? 3) How do such behaviors pose a threat to Internet users? To answer these questions, we analyzed DNS responses obtained using an open resolver probing system, which we describe in the following.

A. Measurement System

The overall flow of open resolver probing is shown in Fig. 2. In this figure, the flow of Q1, Q2, R1, and R2 corresponds to the DNS query from the prober to the open resolver, the DNS query from the open resolver to the authoritative name server, the DNS response from the authoritative name server, and the DNS response from the open resolver to the prober, respectively. The root name server and the TLD name server are not shown in this figure because they are out of the scope of this study. The communication with both servers takes place in the time between Q1 and Q2 to find the address of the authoritative name server. To gather all flows in Fig. 2 during our measurements, we built and controlled two components: a prober and an authoritative name server. In the following, we elaborate on the details of each component.

1) Open Resolver Prober: A prober is responsible for sending 'A' type queries to the entire IPv4 address space (Q1) and collecting responses from open resolvers (R2). The Q1 messages generated by a prober include the subdomains underneath *ucfsealresearch.net*, which is under our management.

a) Probing system: To perform DNS probing, we modified ZMap [17], an open-source fast Internet-wide scanner. In theory, ZMap is able to probe the entire IPv4 address space within an hour. To cope with our limited bandwidth, I/O constraints, etc., we performed a probing at 100k packets-per-second (pps). We implemented a prober by combining the latest ZMap with the subdomain generation in section III-B.

b) Probing range: In order to capture a snapshot of open resolvers on the Internet, we probed all IPv4 addresses except for some reserved areas (e.g., private network addresses). As a result, a scan of about 3.7 billion addresses was conducted, which resulted in a comprehensive view of open resolvers.

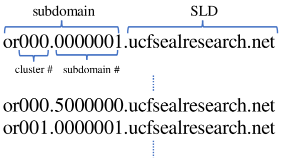

subdomain
SLD
or000.0000001.ucfsealresearch.net
cluster #
subdomain #
or000.5000000.ucfsealresearch.net
or001.0000001.ucfsealresearch.net

Fig. 3. The subdomain structure for open resolver probing.

2) Authoritative Name Server: Upon receiving a DNS query, the open resolver starts a recursive resolution. The interpretation of the domain name proceeds in the order as shown in Fig. 1. Among the components that make up the whole DNS, it is impossible to build a root name server or a TLD server by ourselves, so we built an authoritative name server to observe the behavior of open resolvers. The authoritative name server is responsible for the translation of subdomains that belonged to the Second-Level Domain (SLD) in the DNS query. A prober generates DNS queries that include our SLD, which makes our authoritative name server participate in the recursive resolution process as a last step (⑥ and ⑦ in Fig. 1), the subdomain translation.

a) Second-level domain: We purchased an SLD, *ucfsealresearch.net*, from GoDaddy [18] to enable the configured authoritative name server to manage the domain name resolution of its subdomains. We set our authoritative name server as the responsible DNS server of the purchased SLD.

B. Subdomain Generation

To understand the behavior of the open resolver, we need to keep track of Q1, Q2, R1, and R2 for each open resolver. Basically, DNS matches the pair of the query and the response using the ID field in the DNS header. However, it is infeasible to assign a unique ID number to each query-response pair in our measurement, because the DNS ID field is only 16 bits which can represent up to 65,535 IDs, while the probing rate is about 100k pps. Therefore, we implemented and applied a subdomain generation method to deal with this issue. During the probing process, DNS queries with different subdomains (e.g., *or000.0000000.ucfsealresearch.net*, *or000.0000001.ucfsealresearch.net,* etc.*)* are sent to different IP addresses. Using the qname information contained both in the DNS request and response, we were able to easily group Q1, Q2, R1, and R2 for each flow.

1) Subdomain Cluster: Considering the limited memory resource of the authoritative name server, it cannot load about 4 billion subdomains for all IP addresses at once. In our authoritative name server, only about 5 million subdomains could be reliably loaded. Therefore, we devised a two-tiered subdomain structure for the measurement as shown in Fig. 3.

We grouped the 5 million subdomains that can be provided at once by the authoritative name server into one cluster. Five million subdomains, where each has a unique number (right 7 digits in the figure), are generated as one cluster (a zone file), and each cluster is numbered (left 3 digits in the figure). Once the predetermined number of subdomains in the cluster is exhausted, the cluster is updated with a new cluster number.

2) Subdomain Reuse: The application of subdomain and clustering allows us to easily match Q1, Q2, R1, and R2 by comparing the qname field in DNS packet as well as to prevent the cached response. However, creating a cluster of 5 million subdomains also increases the probing time. To be specific, it takes about one minute to load 5 million subdomains at the authoritative name server, and the time will

TABLE I

THE SUMMARY OF THE OPEN RESOLVER PROBING. NOTICE THAT THE NUMBERS IN PARENTHESES IN THE Q2 AND R2 SHOW THE PERCENTAGE OF EACH NUMBER TO THE NUMBER OF Q1

Start time	End time	Duration	Q1	Q2, R1 (%)	R2 (%)
10/28/2013 2PM	11/04/2013 6PM	7d 5h	3,676,724,690	38,079,578 (1.0357)	16,660,123 (0.453)
04/26/2018 3PM	04/27/2018 2AM	11h	3,702,258,432	13,049,863 (0.3525)	6,506,258 (0.1757)

TABLE II

THE PRESENCE AND CORRECTNESS OF dns_answer FIELD IN R2. NOTICE THAT W AND W/O CORRESPOND TO THE NUMBER OF R2 PACKETS WITH AND WITHOUT dns_answer, RESPECTIVELY. W_{Corr} AND W_{Incorr} CORRESPOND TO THE NUMBER OF CORRECT AND INCORRECT ANSWERS, WHICH RESULTS IN $W_{Corr} + W_{Incorr} = W$, AND Err MEANS THE PERCENTAGE OF INCORRECT ANSWERS TO THE W, SUCH THAT $Err = W_{Incorr}/W \times 100$

Year	R2	W/O	W		$Err(\%)$
			W_{Corr}	W_{Incorr}	
2013	16,660,123	4,867,241	11,792,882		1.03
			11,671,589	121,293	
2018	6,506,258	3,642,109	2,863,655		3.88
			2,752,562	111,093	

be very long considering that 4 billion IP addresses can make up to 800 clusters in total. As such, we added a subdomain reuse method to improve the performance. The prober parses the response packet (R2) after sending the packets including subdomains within one cluster and reuses the subdomain not in the collected R2 set, indicating that the packet was sent to the IP address, which is not an open resolver. Using this approach, we could reduce the number of clusters for probing from the theoretical value of 800 to only 4.

3) Alternatives: Our system for dynamically generating subdomains at the domain name server can be replaced with DNS software (e.g., PowerDNS). However, at the time of conducting the measurements (both in 2013 and 2018) we were unaware of such software, necessitating the above system.

IV. MEASUREMENT RESULTS

We successfully performed an Internet-wide probing that lasted approximately 10 hours and 35 minutes, where about 3.7 billion Q1, 13 million of each Q2 and R1, and 6.5 million R2 packets were captured at either the prober or the authoritative name server. Compared to the number of Q1, the number of Q2 and R1 is about 0.353% and those of R2 are only about 0.176%. Table I shows a summary of the probing results.

We compare this result with results obtained from a dataset collected in 2013. In 2013, we performed an Internet-wide measurement using a C-based system, not based on ZMap as in this study, which does not affect the settings. As shown in Table I, the probing took about 7 days for sending about 3.7 billion Q1. We collected about 38 million Q2 and R1, and about 16.6 million R2 packets. The percentages of Q2 (R1) and R2 to the number of Q1 are about 1.0357 and 0.453, respectively. By observing the reduction of Q1 and R2 counts, we deduce that the number of open resolvers has declined over five years. In the following, we explore the change in the number of open resolvers and their behaviors in-depth.

A. R2 With Empty Question Field

In the sequel, we focus on R2 to analyze the behavior of the open resolvers. However, we remark that some of the collected R2 packets had an empty dns_question field. In general, the dns_question field is included in both the DNS query and the response [19]. As described in section III-B, dns_question is used to group the set of Q1, Q2, R1, and R2. Accordingly, we excluded those 494 packets without dns_question field from our analysis in 2018. As a result, the following analysis only covered 6,505,764 R2 packets with dns_question field. However, we briefly provide a summary of those excluded packets in Appendix A.

1) DNS Answer and Correctness: In this section, we describe a high-level analysis of the collected R2 packets, both presence and correctness. The presence of packets is simply measured by counting the number of R2 at the prober, while the correctness is measured by comparing the translated result in R2 with the ground truth.

As shown in Table II, we observed 16,660,123 R2 packets during the probing in 2013. Out of all R2 responses, 4,867,241 responses do not include dns_answer, while 11,792,882 packets contain dns_answer. Among the R2 packets which have dns_answer field, 11,671,589 packets indicate the correct IP address, but 121,293 responses include incorrect information. The rate of incorrect information is about 1.029%.

In 2018, on the other hand, we found that 2,863,655 DNS responses out of total 6,505,764 R2 collected packets had dns_answer, while the remaining 3,642,109 packets do not. Moreover, 2,752,562 of the 2,863,655 dns_answer fields contained the correct IP address result, and the other 111,093 responses had wrong results (3.879%).

From this result, we can conclude that the number of R2 packets with dns_answer has greatly decreased from about 11.8 million to 2.9 million. The reduction (≈9 million) is similar to the reduction in the R2 packets (≈10 million). Interestingly, however, the number of R2 packets providing misleading information remains similar (≈110 thousand). Consequently, the error rate has increased from about 1% to 4%. From this result, we can infer that the number of resolvers exhibiting unusual behaviors did not significantly change, despite a significant reduction in the total number of open resolvers.

2) Analysis of DNS Header: The operation of DNS is mainly based on RFC1034 [20] and RFC1035 [19]. These documents elaborate the standards for DNS, such as DNS packet header structure as well as the process of DNS resolution. Considering that the open resolver is a component of the whole DNS, we expect it to follow the standard when participating in the translation process.

Through this measurement, however, we found that many resolvers don't follow the standard. To be specific, when the resolvers generate DNS answer packets, many of them fill the DNS flags and the response code fields not according to the instructions described in the standard. In the following, we investigate such behaviors of open resolvers by analyzing the collected data through a comprehensive measurement.

a) Recursion available flag: The Recursion Desired (RD) flag bit in the header of DNS queries sent during the probing is '1', which means that the recursive resolution is required. If the recipient of this query can

TABLE III

THE STATISTICS OF THE dns_answer FIELD AND THE VALUE OF RA BIT IN R2. NOTICE THAT RA$_0$ AND RA$_1$ CORRESPOND TO THE VALUE OF RA FLAG BIT

	2013					2018				
	W/O	W		Total	$Err(\%)$	W/O	W		Total	$Err(\%)$
		W_{Corr}	W_{Incorr}				W_{Corr}	W_{Incorr}		
RA$_0$	4,147,838	241,950		4,389,788	31.35	3,434,415	69,166		3,503,581	94.23
		166,108	75,842				3,994	65,172		
RA$_1$	719,403	11,550,932		12,270,335	0.39	207,694	2,794,489		3,002,183	1.64
		11,505,481	45,451				2,748,568	45,921		

perform recursive resolution (open resolver), it proceeds with recursion on behalf of the prober. Once the open resolver knows the result, it returns the translated IP address with the Recursion Available (RA) bit of ‘1’ to the prober.

In this work, the investigation of the RA flag in R2 started to figure out how many open resolvers exist. According to the definition of open resolver, a publicly accessible and recursion-available resolver, we expected the open resolvers to answer our query with the RA bit of 1 and a correct answer. However, by looking into the collected data, we found that RA bit does not directly mean the existence of an open resolver.

Table III shows the analysis of RA bit in R2 packets. In 2013, we can see that the RA bit of 12,270,335 R2 packets appeared as 1, which might imply that there were about 12 million open resolvers. The interesting observation we make is that there were 241,950 DNS responses with dns_answer field, even though they also have the RA bit of 0 (recursion unavailable). Moreover, 166,108 of them include the correct IP address information, which means that the senders of those 166,108 packets actually play the role of the open resolver, although they indicate they are not open resolvers. Conversely, there were 719,403 DNS responses without dns_answer field, but with the RA bit of 1 (recursion available), which means they do not perform the resolution. Such resolvers can be assumed to be either publicly inaccessible or unable to perform the recursive resolution. If the latter is the reason for the blank answer, it can be inferred that those resolvers do not follow the standard implementation for DNS resolution.

In the result collected in 2018, 3,503,581 R2 packets have the RA bit of 0 (recursion unavailable), while 3,002,183 include RA bit of 1 (recursion available). Moreover, among the responses with RA bit of 0, 69,166 packets include dns_answer, 3,994 of correct answers and 65,172 of incorrect answers, which results in an error rate of 94%. On the other hand, 207,694 R2 responses do not contain any resolved IP address, even though they claim to have recursion available.

Regardless of the value of the RA bit, the number of packets with the dns_answer field decreased to about one quarter (from 241 thousand to 69 thousand for RA bit of 0 and from 11.5 million to 2.8 million for RA bit of 1). However, the number of incorrect answers is similar, or even larger in 2018 (from 75 thousand to 65 thousand for RA bit of 0 and from 42 thousand to 46 thousand for RA bit of 1).

In terms of the accuracy of the response, when the RA bit is 0 and the dns_answer field is included, the resolved IP address is often wrong in 2018, with 94.225% of wrong dns_answer fields. Moreover, 31.346% of responses with RA bit of 0 and dns_answer field include incorrect information in 2013. If the RA bit of 0 and dns_answer were given together, the probability that the included IP address was inaccurately increased by more than three times. When the RA bit is 1, it can be seen that about 0.393% and 1.643% of the packets containing dns_answer include the wrong result in 2013 and 2018, respectively. Considering that only less than 6% of the cases with the RA bit of 0 include dns_answer field, the ratio of incorrect responses to the total would be low. Obviously, however, an improper combination of the RA bit and dns_answer can be a clear indicator of a false result.

While investigating the RA bits in the responses, it is difficult to determine the number of open resolvers. It can be estimated that there were about 11.5 million open resolvers with the strictest criteria in 2013 (with the RA flag of 1 and correct dns_answer). Using the same criteria, it is estimated that there are about 2.74 million open resolvers in 2018. However, if we only use the RA flag as a criterion, we can also estimate that there were 12.2 and 3 million open resolvers in 2013 and 2018, respectively. On the other hand, it is also possible to conclude that there were about 11.7 and 2.75 million open resolvers by only counting the R2 packets with correct answer regardless of the RA bit.

b) Authoritative answer flag: Authoritative Answer (AA) means that the server responding to the DNS query is the authoritative name server for the given domain. In our probing, we sent DNS queries for all IPv4 addresses, and we were the only owner for the SLD, *ucfsealresearch.net*. Therefore, intuitively, it is appropriate to assume that the AA bit of all R2 is set to 0 except one response from our authoritative name server itself. However, as shown in Table IV, 381,124 R2 packets came back with the AA bit of 1, which is about 2.29% of all responses in 2013. Among them, 231,368 responses contained dns_answer, of which 78,279 had incorrect results. The ratio of the false answers to total answers with the AA bit of 1 is about 20.539%, which is significantly high compared to 0.372% when the AA is 0.

In 2018, we received 249,193 R2 responses with AA of 1. Among them, 119,147 packets had dns_answer and 94,052 had incorrect information (79%), which is more than twice the rate of 2013, while the rate for AA bit of 0 is about 0.621%. The number of responses with AA bit of 1 is less than 4% of the total answers, while incorrect answers with AA bit of 1 account for about 84.661% of all incorrect packets.

Comparing 2013 and 2018's results, we see that the R2 with AA bit of 0 is greatly reduced (from 16 million to 6 million). In the case of AA bit of 1, the number of packets decreased from about 381k to 249k, which is about 61%. Nevertheless, the value of 249k is still abnormally higher than the expected value of 1. As interesting as the findings of the RA bit, however, the number of responses with inaccurate information was similar or even higher, which results in a significantly high error rate in 2018, which more than doubled from 2013.

As in the previous analysis of the RA flag, the analysis of the AA flag also shows that a large number of open resolvers do not work in a reliable manner. Although that might not be as surprising, since they are ‘open’ (out of recommendation), we still want to highlight the results for the following reasons:

TABLE IV

THE STATISTICS OF THE dns_answer FIELD AND THE VALUE OF AA BIT IN R2. NOTICE THAT $\texttt{AA}_0$ AND $\texttt{AA}_1$ CORRESPOND TO THE VALUE OF AA FLAG BIT

	2013						2018					
	W/O	W			Total	$Err(\%)$	W/O	W			Total	$Err(\%)$
			W_{Corr}	W_{Incorr}					W_{Corr}	W_{Incorr}		
$\texttt{AA}_0$	4,717,485	11,561,514	11,518,500	43,014	16,278,999	0.37	3,512,053	2,744,518	2,727,477	17,041	6,256,571	0.62
$\texttt{AA}_1$	149,756	231,368	153,089	78,279	381,124	20.54	130,046	119,147	25,095	94,052	249,193	78.94

TABLE V

THE RCODE OF THE DNS REPONSES. NOTICE THAT EACH COLUMN CORRESPONDS TO RCODE OF 0 TO 7, AND 9. THE RCODE OF 8 (NXRRSET) IS OMITTED DUE TO THE ABSENCE IN OUR DATASET

		NoError	FormErr	ServFail	NXDomain	NotImp	Refused	YXDomain	YXRRSet	Not Auth
2013	W	11,780,575	0	12,723	10	0	1,272	0	0	0
	W/O	1,198,772	453	354,176	145,724	38	3,168,053	0	2	11
	Total	12,979,347	453	366,899	145,734	38	3,169,325	0	2	11
2018	W	2,860,940	23	2,489	10	0	193	0	0	0
	W/O	377,803	233	200,320	48,830	605	2,934,269	1	2	80,032
	Total	3,238,743	256	202,809	48,840	605	2,934,462	1	2	80,032

1) conversely, we can still find many open resolvers working in harmony with the standard, and 2) through this study we aim to empirically analyze how these abnormal behaviors relate to their maliciousness.

c) Response code: The response code (rcode) of the DNS response provides metadata about the outcome of the resolution. The rcode, which usually has a value from 0 to 15, is set to 0 for NoError, 1 for FormErr, 2 for ServFail, 3 for NXDomain, 4 for NotImp, 5 for Refused, 6 for YXDomain, 7 for YXRRSet, 8 for NXRRSet, and 9 for NotAuth [21]. All but 0 (NoError) indicate that the resolution was not successful.

Table V shows the distribution of rcode in the collected R2. As expected, most responses containing dns_answer field contained an rcode of 0, while most responses had a nonzero rcode without dns_answer. Except for this general tendency, however, we found some R2 packets with abnormal combinations: 14,005 contained a nonzero (error) rcode despite having dns_answer field; 12,723 for ServFail, 10 for NXDomain, and 1,272 for Refused. Conversely, 1,198,772 R2 packets without the dns_answer field had rcode of NoError.

In 2018, we found 2,715 R2 packets with a nonzero rcode and dns_answer field. In particular, 23 R2 s have rcode of 1, 2,489 have 2, 10 have 3, and 193 have 5; 377,803 responses with rcode of 0 had no dns_answer field. In analyzing response code, we found that the number of packets with most response codes decreased (NoError, FormErr, ServFail, NXDomain, and Refused). However, we also can see that the number of responses with the rcode of 1 (NotImp) and 9 (Not Auth) significantly increased, while those with the rcode of 6 (YXDomain) and 7 (YXRRSet) remained at a similar level.

B. Remarks

While we observed many DNS responses do not follow the standard, however, we note that not all abnormal responses will trigger an error on the prober-side. For instance, we observed 249,193 resolvers responded to our query with *AA* flag bit masked as ‘1’, which implies that they attempt to disguise themselves as an authoritative name server. In our measurement, we can easily verify the authenticity of responses, because we have our own authoritative name server. In other words, this type of abnormal response cannot be noticed by clients, as they are not able to verify whether the source of responses is actually the authoritative name server. On the contrary, the most misconfigurations of *RA* and rcode flags can easily be noticed by clients because of the inconsistency between DNS header and body. Therefore, DNS clients must have an error check function to cope with such threats, otherwise, adversaries can easily abuse the flaw to launch DNS hijacking attacks.

TABLE VI

THE SUMMARY OF INCORRECT ANSWERS. NOTICE THAT $\#_{R2}$ MEANS THE NUMBER OF R2 PACKETS THAT INCLUDE THE ANSWER IN EACH FORM, AND $\#_u$ MEANS THE NUMBER OF UNIQUE VALUES APPEARING IN $\#_{R2}$

Form	2013		2018		Example
	$\#_{R2}$	$\#_u$	$\#_{R2}$	$\#_u$	
IP	112,270	28,443	110,790	15,022	216.194.64.193
domain	249	175	231	80	u.dcoin.co
string	10	57	72	29	wild, OK, ff
N/A	8,764	-	-	-	<0x00>
Total	121,293	28,675	111,093	15,131	-

1) Incorrect DNS Answers: In the following, we describe further analysis on the incorrect IP addresses included in R2 packets based on the result observed in 2013 and 2018. As shown in Table II, we notice that the wrong answer was provided in 110,093 packets out of 6,506,258 R2 packets in 2018, while 121,293 packets out of 16,660,123 provided the wrond answer in 2013.

Table VI shows a summary of the incorrect answers collected through the measurement. We categorized 121,293 and 111,093 R2 packets in 2013 and 2018 with the wrong result into three types: IP address, other domain name, string, according to dns_answer. As a result, we found that 112,270 in 2013 and 110,790 in 2018 of the R2 packets had incorrect IP addresses, while 249 and 231 R2 packets had incorrect domain names in their dns_answer fields. We also found that 10 and 72 responses include the abnormal strings such as *wild, ff, OK, 04b400000000, etc.*

TABLE VII

TOP 10 IP ADDRESSES INCLUDED IN INCORRECT DNS RESPONSES IN 2018. 'REPORTS' IS WHETHER A SUSPICIOUS REPORT WAS FOUND WHEN QUERYING THE ADDRESS USING CYMON API

IP address	#	Org Name	Reports
216.194.64.193	23,692	Tera-byte Dot Com	N
74.220.199.15	13,369	Unified Layer	**Y**
208.91.197.91	8,239	Confluence Network Inc	**Y**
141.8.225.68	1,197	Rook Media GmbH	**Y**
192.168.1.1	1,014	private network	N/A
192.168.2.1	741	private network	N/A
114.44.34.86	734	Chunghwa Telecom	N
172.30.1.254	607	private network	N/A
10.0.0.1	548	private network	N/A
118.166.1.6	528	Chunghwa Telecom	N
Total	50,669	-	-

C. Caveats

As mentioned earlier, we used a C based system in the process of collecting dataset in 2013 and stored the results in a.pcap file. While parsing the packets using the libpcap based code, we found in some packets that `dns_answer` was not decoded appropriately. It appears that the open resolver incorrectly filled some values in the process of creating the response packet. Among the total 16 million of R2 packets in 2013, 8,764 packets were not decoded correctly, corresponding to about 0.05% of total number.

a) Top 10 analysis: Table VII shows the top 10 IP addresses with the most occurrences in R2 packets in 2018. The most frequently observed IP address in incorrect `dns_answer` was found in 23,692 responses, which is a domain and web hosting related company. The summed number of top 10 appearances is 50,669, which is about half the total number of incorrect R2 responses (111,093).

By examining the top 10 addresses, we found that four of them are private networks that belonged to 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16. However, in the case of the IP addresses, we found that 74.220.199.15, 208.91.197.91, and 141.8.225.68 are located at the second, third, and fourth places in the table, with suspicious information was found from security information vendor. For IP address 208.91.197.91, for instance, Ransomware Tracker [22] states that the address is a ransomware IP, and Cymon [23] shows that malware, phishing, and botnet activities are reported for the given address as shown in Appendix B. Therefore, 22,805 R2 packets pointing to the IP addresses can be considered to have a deceptive `dns_answer` for malicious purposes.

For the 2013 dataset, we inspected the top-10 frequent IP addresses. The total number of R2 packets that include those addresses is 26,514, which is almost half of the number in 2018. Specifically, in 2013, the most frequently appeared address with 9,651 R2 packets is 74.220.199.15, the second rank in 2018, and it is the only address reported as malicious. Moreover, there are 3 private network addresses, 192.168.1.254, 192.168.2.1, and 192.168.1.1 as a second, third, and tenth places. More than 5k packets, in third place, include the address 20.20.20.20, which is owned by Microsoft, while 173.192.59.63 appeared in 995 packets (seventh rank), 221.238.203.46 in 811 packets (eighth rank), and 68.87.91.199 in 748 packets (ninth rank). As for the unusual point, 1,032 packets include 0.0.0.0.

b) Suspicious IP addresses: Based on the possibility of malicious activities performed by open resolvers, we conducted a deeper analysis to identify the open resolvers misleading users to malicious destination. For answers of IP addresses in Table VI, we conducted an additional analysis using Cymon API [24]. From Cymon, we gathered reported information about the given addresses and judged their maliciousness. As a result, we found that there were 335 IP addresses reported as malicious. When there are multiple reports for different categories, the most frequently reported category is selected.

As shown at the right side of Table VIII, the most common category for the malicious IP address is malware. The number of IP addresses related to malware is 170, accounting for over half. Moreover, the number of IP addresses related to phishing is 125, accounting for more than one third, alluding to the possibilities of DNS poisoning or manipulation. Moreover, the number of IP addresses reported as spam, SSH bruteforce, scan, and botnet is about 40. When the analysis is conducted w.r.t. the number of R2 packets, the result is different. In R2 packets, malware addresses account for more than 85% of the total, which means that 170 malware reported addresses are observed in R2, on average 136 times each. On the other hand, the 125 phishing related IP addresses are observed in 2,878 R2 packets($\approx$10% of the total; 23 occurrences for each address).

To measure changes in the malicious use of open resolvers, we also conducted the same analysis on the result in 2013. In total, there were 100 unique malicious IP addresses in 12,874 responses. Among them, 65 addresses appearing in 11,149 R2 packets were reported as malware. For addresses reported as phishing, there were 18 unique addresses that were included in 1,092 responses. In addition to the above two categories, 16 IP addresses in 633 responses were reported as Spam, SSH Bruteforce, Scan, Botnet, and Email Bruteforce.

The interesting observation we make by comparing the results of 2013 and 2018 is that the malicious behavior of open resolvers has increased from 12,874 to 26,926 in terms of the number of R2 responses. By comparing the IP addresses of the resolvers providing malicious responses, we found that 5,176 resolvers misled users in both 2013 and 2018. The number of such resolvers that presented over a long period of time is about 40.2% of the total number in 2013. In 2018, these malicious resolvers accounted for about 19.2% of the total, with the remaining 80.8% newly emerging. This corresponds to more than 100% of increase. From the point of view of the unique IP addresses in the R2 packets, the increase in malicious behavior is also significant: from 100 unique addresses to 335 addresses, which is more than tripled (235%).

The number of unique IP addresses reported as malware has increased from 65 to 170, but the ratio to all malicious addresses decreased from 65% to 50%. The most rapid change can be found in phishing: from 19 in 2013 to 125 in 2018, which is about seven folds increase. The ratio has also doubled from about 19% to 37%, indicating that 2018's open resolvers are more exploitable for phishing purposes than before.

Our analysis is considered a lower bound of the malicious activities because it only deals with information in Cymon. However, more malicious addresses may appear when validating using threat information from multiple vendors.

D. Queries With Known Domains

To investigate the behavior deeper, we considered querying known domains as of Jan. 2021. We randomly selected 100 resolvers that provided the malicious responses in 2018 and sent the 5 queries with well known domain names

TABLE VIII

MALICIOUS IP ADDRESSES IN R2 PACKETS. NOTICE THAT $\#_{IP}$ CORRESPONDS TO THE NUMBER OF IP ADDRESSES REPORTED TO CYMON IN EACH CATEGORY. WHEN THE IP ADDRESS IS REPORTED WITH MULTIPLE CATEGORIES, THE CATEGORY WITH THE MOST FREQUENCY IS SELECTED. NOTICE THAT $\#_{R2}$ MEANS THE NUMBER OF R2 PACKETS THAT INCLUDE THE IP ADDRESSES BELONGED TO EACH CATEGORY

Report Category	2013				2018			
	$\#_{IP}$	$(\%_{IP})$	$\#_{R2}$	$(\%_{R2})$	$\#_{IP}$	$(\%_{IP})$	$\#_{R2}$	$(\%_{R2})$
Malware	65	65.0	11,149	86.6	170	50.7	23,189	86.1
Phishing	19	19.0	1,092	8.5	125	37.3	2,878	10.7
Spam	4	4.0	67	0.5	15	4.5	44	0.2
SSH Bruteforce	2	2.0	2	0	10	3.0	323	1.2
Scan	8	8.0	493	3.8	9	2.7	388	1.4
Botnet	1	1.0	70	0.5	4	1.2	102	0.4
Email Bruteforce	1	1.0	1	0	2	0.6	2	0
Total	100	-	12,874	-	335	-	26,926	-

(e.g., google.com, amazon.com, *etc*). As a result, we could see that 37 resolvers, as of Jan. 2021, refused our queries, 29 resolvers timed-out, and 34 resolvers provided us consistent incorrect (malicious) responses in the `dns_answer` fields. This further suggests consistency in their behavior, with no correct answers.

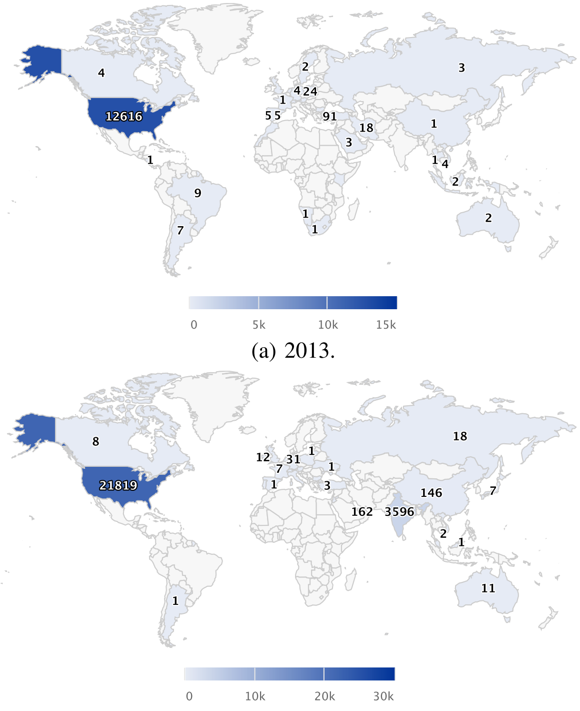
E. Distribution of Malicious Resolvers

To further explore malicious resolvers, we resolve their IP's geolocation and the autonomous system (AS) using ip2location [25]. The IP mapping was done in 2013 and 2018, for accurately pinpointing the location at the time of the survey, and to cope with dynamics. Similarly, note that whether an address is malicious or not is determined at the time of data collection, not analysis.

As a result of this mapping, we found that 12,874 malicious resolvers in 2013 were distributed over 36 countries, with 98% of them in the US and the rest spread over the world. On the other hand, the 2018 analysis unveiled that 21,819 out of 26,926 (about 81%) resolvers were located in the US, followed by 3,596 in India, 714 in Hong Kong, 291 in British Virgin Islands, 162 in United Arab Emirates, and 146 in China, which had a few addresses in 2013. A complete depiction of the distribution is shown in Figure 4. From this analysis, we conclude that the distribution of the open resolvers has been widening (shifting) spatially over time, which will perhaps lead to more impact of those open resolvers if malicious (as they are likely to be used by users within proximity of them). Moreover, we found that the number of malicious resolvers is not always in sync with the total number of resolvers associated with the country: counter-examples include United Arab Emirates, Hong Kong and Virgin Islands, which had a higher number of malicious addresses in 2018, compared to their corresponding address allocations [26]. However, one possible explanation for this bias is that those countries are known as Internet and business hops, making them a good target for malicious behaviors.

F. DNS Manipulation

The above analysis shows that DNS manipulation happens. Queries sent to each IP address were a subdomain instantaneously created, and subsequently manipulated. As mentioned earlier, one of the purposes of using subdomain is to prevent caching of results at the open resolver. In other words, the malicious IP address in the `R2` packets we received does not match the information stored in the cache of the open resolver, but it is likely to be the result of an actual but illegitimate response. It is unreasonable to assume that an attacker applies a cache poisoning to the legitimate open resolver, because of the short time window, but it is more plausible to say that the open resolver itself is under the adversary's control. It can be assumed that those open resolvers will work in a way that provides the predetermined answer, which includes the malicious IP address for every query they receive.



(a) 2013.

(b) 2018.

Fig. 4. Distribution analysis results: 2013 vs. 2018.

1) DNS Header in Malicious Responses: In addition to the general analysis, we also provide an analysis of `R2` packets that may mislead the users to malicious IP addresses.

G. RA and AA Flags

Table IX shows the statistics of `RA` and `AA` flags in `R2` packets that contain a malicious IP address. With regard to the `RA` bit, more than 70% of `R2` packets indicate that the senders are recursion unavailable although the responses have the `dns_answer` field. On the other hand, about 27% of `R2` packets include the `RA` bit of 1, which means that the contained `dns_answer` fields are the result from recursive resolution. However, we already know that the IP addresses in those `R2` packets are malicious and not true, which allows us to infer that `RA` bit is used improperly.

TABLE IX

RA and AA Analysis on R2 Packets With the Malicious IP Address in 2018. Notice That $\#_R$ and $\#_A$ Correspond to the Number of Packets With Each Flag and Value. Also, $\%_R$ and $\%_A$ Correspond to the Percentage of Each Flag to the Total R2 Packets Including the Malicious Information (26,926)

RA	$\#_R$	$\%_R$	AA	$\#_A$	$\%_A$
$\mathtt{RA}_0$	19,534	72.5	$\mathtt{AA}_0$	7,472	27.8
$\mathtt{RA}_1$	7,392	27.5	$\mathtt{AA}_1$	19,454	72.2

TABLE X

Examples of Duplicated Queries in R1 Packets From Different IP Addresses. The Subdomain (or000.2543237) Was Originally Sent to and Returned by 5.188.178.199

Subdomain	IP (Block)	#	Organization
or000.2543237	74.125.0.0/16	72	Google LLC.
	173.194.97.0/25	14	Google LLC.
	162.209.124.87	1	Rackspace Hosting
	200.32.248.1	1	Belize Telemedia LTD.
	200.32.218.132	1	Belize Telemedia LTD.

We also make several interesting observations from the AA bit in R2 packets. More than 70% of the responses have a AA bit of 1, which means that they are from the authoritative name server. Considering that they were not directly sent to our authoritative name server, and even they included the malicious IP address and not true result, the use of AA flag can be assumed to be a malicious attempt to allude to the credibility of the response.

H. Response Code

In the analysis of rcode, we found that all 26,926 R2 packets with malicious IP addresses have the rcode of 0 (NoError). The use of rcode can also be seen as an intention to encourage the requester to trust the response and to access the IP address by claiming a reliability of the answer.

V. Analysis of Packets at Authoritative Server

In this section, we analyze the Q2 and R1 packets captured at the authoritative name server in 2018. By incorporating the analysis of the packets at the authoritative name server and prober, we characterize open resolver behaviors in detail.

A. Q2 *and* R1 *Analysis*

As in Table I, we captured about 13 million Q2 and R1 packets at the authoritative name server. Only about half of them were delivered to our prober back as R2 responses. Thus, we do further exploration to understand the rest of the packets.

1) Summary: First, we describe a summary of Q2 and R1 packets collected at the authoritative name server. As we described earlier, Q2 and R1 packets at the authoritative name server were captured using tcpdump as pcap files. In order to analyze them in detail, we implemented a C-based parser using libpcap and looked into the collected DNS packets. Since our main goal is to understand how the prober's query for subdomains would be handled by open resolvers and the authoritative name server, we excluded 'non-A' type (e.g., NS, CNAME, *etc*) packets from about 13 million collected packets. As a result, we are left with 10,638,510 R1 packets as our dataset. In the meanwhile, we observed that the number of packets in Q2 and R1 differs slightly, but this is probably due to missing packets by tcpdump or duplicated queries, which will discuss further below. In essence, the authoritative name server is under our control, so we safely assume no malicious behavior by the authoritative, such as packet tampering. Hereafter, we focus only on R1 packets in the following analysis.

2) Duplicated Queries: The next step is to check the duplicated queries. As we explained in section III-B, for probing, we generated a unique subdomain for each target address, which means no DNS queries share the same `dns_question` field. However, in our dataset, we found that there exists such a significant number of duplicates. For example, the authoritative name server received the queries about one subdomain, or000.2543237.ucfsealresearch.net, from 89 different IP addresses. As shown in Table X, 72 addresses belonged to 74.125.0.0/16 and 14 addresses belonged to 173.194.97.0/25, while both subnets are owned by Google LLC. On the other hand, there are three R1 packets from addresses not by Google.

An interesting observation we make is that the Q1 for this subdomain was originally sent to 5.188.178.199, which is owned by Fast Content Delivery LTD. This address does not appear to perform recursive resolution directly for the query we sent, but it tries to utilize Google's or another organization's DNS server as forwarders to translate the given subdomain. However, what is even more interesting is that although the given subdomain was translated by our authoritative name server along with the other DNS server, such as Google, and the result was highly likely to be returned to the original resolver, the resolver did not deliver the correct answer to the prober. The R2 response from 5.188.178.199 does not include a `dns_answer` field in the packet but includes an rcode of 5 (Refused). In summary, the DNS server refused the user's (prober's) query but performed resolution through other DNS servers (forwarders) for unknown reasons. Analysis of whether a recursive resolution is actually performed and the accuracy of R2 packets is discussed in detail later.

Overall, we investigated the subdomains in 10,638,510 R1 packets and found that R1 packets for 35,320 subdomains were returned to two or more different resolvers or forwarders. Moreover, there are 6,423,321 unique domain names, while 4,215,189 subdomains are included in two or more queries.

3) Duplicated IP Addresses (Forwarder): While examining the R1 packets, we observe that there is a relatively small number of unique IP addresses. Namely, we found that, while we had about 10.6 million R1 packets, the number of unique IP addresses in our dataset is only 133,401. Compared to the 6,505,764 R2 responses all having different IP addresses, this finding in itself is very interesting and highlights that the number of resolvers that directly sent Q2 to our authoritative name server is very small. While the existence and use of forwarders in the DNS ecosystem are well-known, inspired by our dataset collected from both authoritative name servers and the prober, we pursued the empirical analysis of forwarders by comparing the captured R1 and R2 packets.

We analyzed those 133,401 unique addresses and found that 81,441 addresses appeared in only one R1 query, which means that they performed recursive resolution only once. In other words, DNS queries for the rest of 10,557,069 R1 packets were sent to the remaining 51,960 IP addresses. Again, considering that the 6.5 million R2 all have different addresses, a resolution process of about 6.42 million was actually performed by DNS servers that sent more than one Q2.

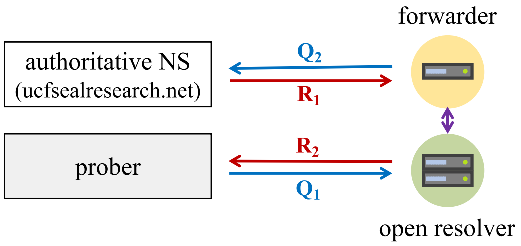


Fig. 5. The flow of DNS request and response packets with a forwarder in the resolution. Unlike Fig. 2, a recursive resolution is processed by the forwarder. Our analysis indicates that those forwarders play an essential role in explaining a large number of incorrect or malicious answers.

TABLE XI

THE TOP 5 FORWARDERS' IP ADDRESSES SENDING THE LARGEST NUMBER OF Q2 PACKETS. NOTICE THAT # MEANS THE NUMBER OF Q2 PACKETS SENT FROM AN IP ADDRESS

IP address	#	Org Name (Country)
41.110.32.2	148,718	Algeria Telecom (Algeria)
41.110.30.2	121,698	Algeria Telecom (Algeria)
156.200.40.50	93,368	Te Data (Egypt)
41.110.31.2	69,729	Algeria Telecom (Algeria)
200.75.51.133	69,501	ETB (Colombia)
Total	503,014	-

From this analysis, we infer the existence of forwarders that do not appear in Fig. 2. As shown in Fig. 5, a forwarder receives a DNS query from another resolver and performs recursive resolution. The existence of such forwarders suggests a new interesting point when analyzing the accuracy and maliciousness of the `dns_answer` fields: A wrong `dns_answer` in `R1` can be manipulated by the forwarder before sending it to the resolver, and further be delivered to the end-users. We will investigate deeper in the following section where the manipulation of the answer occurs.

Table XI shows the top 5 forwarders sending the largest number of `Q2` packets to the authoritative name server. Interestingly, three of them are owned by an Algerian company, namely Algeria Telecom. The total number of `Q2` sent from these top five addresses totaled over half a million.

B. Comparing `R1` *and* `R2`

We also reexamined the accuracy of `R2` responses. Instead of simply verifying the resolved IP address in the `R2` responses as in section IV-A.1, we observed the behavior of open resolvers in conjunction with the collected `R1` packets.

1) Direct Resolution (Without Forwarder): As we discussed above, in many recursive resolution processes, a forwarder was included to deliver the DNS query and response between the prober and the authoritative name server. As such, our first concern is the number of open resolvers that actually perform recursive resolution without forwarders. To figure them out, we examined the number of resolvers' IP addresses that appeared in the {`R1`, `R2`} pair. As a result, we found 70,694 open resolvers (about 1.09% of collected `R2` packets) that performed direct recursive resolution without employing a forwarder. This number is less than the 81,441 addresses that only sent a query for one subdomain, discussed in the previous section. In other words, among the 81,441 IP addresses that sent only one query, the number of open resolvers that performed a direct resolution is 70,694. The remaining 10,747 IP addresses can be considered as forwarders that participate in only one DNS resolution. It can also be inferred that those forwarders are only used in a few limited cases, and are not available publicly.

TABLE XII

THE NUMBER OF PACKETS IN R1, R2, THEIR INTERSECTION, AND DIFFERENCES. NOTE THAT R1- R2 MEAN THAT THE NUMBER OF PACKETS INCLUDED IN THE R1 DATASET BUT NOT IN THE R2

	#	R1 ∩ R2	R1- R2	R2- R1
R1	10,638,510	3,215,947	5,546,983	3,289,817
R2	6,505,764			

2) Subdomain Based Analysis and Correctness: Next, we examined the correctness of the resolution results by looking into each {`R1`, `R2`} pair. The reason for this analysis is to further explore the misbehavior and malfunction of the forwarders and resolvers, such as cases that result in empty `dns_answer`, packet drop, manipulation, *etc*.

In general, the process of domain name resolution by the open resolvers follows Fig. 1 (regardless of the existence of the forwarder in Fig. 2 and Fig. 5). As a result, if the translation of the subdomain is done successfully, the packet flow must appear in both `R1` (`Q2`) and `R2` at the same time. In particular, in our experiments, we used different subdomains to eliminate the effect of caching on our visibility into the resolution from the open resolvers. As such, we can assume that all resolutions will follow the flow. However, we observed two abnormal cases: 1) packets that exist in `R1` but not in `R2` and 2) packets that exist in `R2` but not in `R1`. Along with these two cases, we also included a general resolution process that goes through both `R1` and `R2`. Notice that we conducted our analysis based on the subdomain, not the IP address, which means the existence of the same IP address (in the case of using a forwarder) does not affect our end results. Table XII shows a summary of the statistics of `R1` and `R2`.

a) Packets in `R1` *but not in* `R2` *:* The first case we considered is packets that exist in `R1` but not in `R2`. This case implies that an authoritative name server received the query and replied back and that the resolution result is discarded or dropped in the middle. We observed that 5,546,983 out of 10,638,510 subdomains in `R1` are not seen in `R2`. In this case, since resolvers did not provide the resolved address to the prober, we cannot verify its accuracy. However, it can be seen that a large number of resolvers exhibit abnormal behavior. Notice that there is also the possibility of an unreliable network, which may result in dropping some packets. However, a large number of dropped packets (more than 50%) alludes to the former explanation, where the dropped packets due to the network condition are often very small.

b) Packets in `R2` *but not in* `R1` *:* The second case is packets that exist in `R2` but are not in `R1`. In this case, it seems weirder than the previous case because it provides the answer without going through the normal resolution process. This case may occur when a resolver returns a predetermined result without recursive resolution. We found that more than half (i.e., 3,289,817) of the total `R2` packets (i.e., 6,505,764) fall into this case. However, among the 3,289,817 `R2` packets, 95.27% of them (i.e., 3,134,036) do not have a `dns_answer` field, which happens when the DNS queries are responded to without contacting the authoritative name server. This number also corresponds to 86.05% of all `R2` responses without `dns_answer` fields in Table II.

On the other hand, there are 155,781 R2 packets that include dns_answer fields although they did not perform resolution. Among them, 95,840 R2 responses included incorrect answers, which happens when the resolver injects the predetermined (or random) result. Besides, we also found that 59,941 R2 packets contain correct responses, while those packets are not in R1 (less than 0.5% of total captured R1 packets). The potential reasons behind this are: 1) missing packets on tcpdump, 2) different caching schemes at the resolvers (e.g., SLD-based caching), and 3) an open resolver equipped with multiple IP interfaces and employs a shared cache. As future work, we will further explore this case of packets and resolvers.

c) Packets in both R1 *and* R2 *:* The third case is the most general case, which is a common recursive resolution process. In other words, the conversion is done via the authoritative name server by the open resolver, and the result is returned back to the prober. In this case, it is expected that the correct resolved address would be included in a dns_answer in the R2 packets. However, we observed that 508,082 R2 packets (about 15.8% of the subdomains) included in both R1 and R2, were returned to the prober without dns_answer fields. For example, the subdomain (or000.2543237) introduced in Table X represents this case. In addition, 2,707,865 R2 packets, which are about 84.2% of this case, were returned with a resolved address, of which 2,692,621 packets delivered the correct answer and 15,244 packets contained the incorrect answer address.

3) Malicious Answers: We also conducted an additional analysis of the malicious answers in Table VIII. This analysis is for figuring out how many of the 26,926 packets that led us to a malicious destination were actually resolved with the authoritative name server and how many of them used forwarders. To do this, we listed the subdomains that have been translated into malicious addresses and looked them up in the R1 dataset.

As a result, we found that 23,349 subdomains out of 26,926 (about 86.72%) in the R2 packets do not appear in the R1 dataset. This finding means that some resolvers did not perform recursive resolution for the domain name queried by the prober, which also means that their responses are highly likely to be predetermined. This includes the possibility that the targeted resolver sent the query to the forwarder, but the forwarder provided a predetermined answer without contacting the authoritative name server. However, since we could not look into the communication between the targeted resolver and forwarder, it is difficult to verify this theory.

The second is the case that open resolvers performed resolution through the authoritative name server, and 3,577 R2 packets (about 13.28%) belonged to this case. Among them, 2,935 resolvers delivered queries to the authoritative name server via forwarders, while 642 sent queries directly to our server without forwarders. Given the high likelihood that a specific malicious destination is likely to be a predetermined value, it is uncertain why they sent a query to the authoritative name server to get a resolved address. However, we can at least see that the resolver itself manipulated the answers in 642 R2 packets, while the answers in 2,935 R2 packets could have been manipulated by either the resolver or the forwarder.

VI. Discussion and Future Works

A. The Need for Continuous Monitoring of Open Resolvers

As show in the above analysis, open DNS resolvers still pose a threat to the Internet. The fact that the number of open resolvers has declined does not mean that their threat is going to go away anytime soon. For example, the number of open resolvers with a malicious behavior has increased, which is a clear example of the need for steady observation of those resolvers and the role they play in the DNS ecosystem.

However, and to the best of our knowledge, such a continuous and steady observation of the open resolvers on the Internet is not well performed. For example, one of the most popular open resolver-related projects is the openresolverproject.org [5], which shows the number of open resolvers distributed over the Internet and some flag values (RA bit or rcode). However, this project but does not provide any in-depth analysis of malicious IP addresses included in the responses. Moreover, and most importantly, *the project has been discontinued since January 2017*.

Another project, which is called the shadowserver dnsscan [27], provides daily information on the number and geographical distribution of open resolvers, but does not specifically analyze the behavior of each open resolver. Therefore, it is difficult to use the result of this project for understanding the threat of open resolvers in such relevant details. This is because the decrease in the number of open resolvers, as pointed by our work, does not directly mean that the associated threats are also reduced.

Censys [28] and Rapid7 [29] provide weekly (censys) or monthly (rapid7) scan using ZMap. These raw scans are useful for DNS response packets can be inspected, but still have limitations. First, this raw dataset is from the measurement result only using prober, not the authoritative name server. As shown in Fig. 2, if the measurement is conducted only at the prober, we cannot catch the packet flow of R1 and Q2, which makes it difficult to investigate the behavior of open resolvers in-depth. Moreover, because both repositories use ZMap as a scanning tool, these datasets may have a blind spot that ZMap has. For example, the current ZMap can miss packets in that it only stores results for the responses from the target port of the scan (e.g., DNS responses only from port 53 or any ICMP packets), while it filters out the DNS packets from other ports. This incomplete measurement can lead to the underestimation of the threat of misbehaving resolvers.

Thus, we believe a systematic and constant follow-up of the behavioral analysis in the open resolver ecosystem is a gap in the literature, and is needed for improving DNS security. For understanding the behavioral changes in open resolvers and finding countermeasures against the malicious activities such a steady observation is required.

B. Private Network in Incorrect Information

In Table VII, we show that four of the top 10 IP addresses with incorrect R2 responses in 2018 are addresses in private networks (196.168.1.1, 192.168.2.1, 172.30.1.254, and 10.0.0.1). Besides the Top 10, several private networks appeared in the incorrect responses as well.

We speculate multiple scenarios that could lead to such a behavior. For example, landing on such a private network may be a redirection to a webpage for the user's consent or form submission in a public network (e.g., Wi-Fi in airport). It is also likely to be a similar redirection in the responses with the particular company's IP address. However, in the case of a private network, and given the fact that our DNS query was sent from outside the network, this behavior is still difficult to understand. If it is a DNS server for users inside the network, it means that the connection is also allowed from the outside.

C. Open Resolver as an Existent Threat

In section II-B, we described two threats that open resolvers can bring about: DNS amplification (DDoS attacks) and DNS manipulation. In our analysis, we found that there are millions of open resolvers still exist in the wild, which allows us to deduce that these resolvers can be exploited by adversaries for launching amplification attacks. The number of open resolvers around the world can be equated to the magnitude of the potential threat as it is a threat from the functional loophole of the open resolver (no verification method for spoofed source IP address is in place). The mere existence of open resolvers and the adversary's malice are a guarantee for an attack.

However, in terms of DNS manipulation, the existence of malicious open DNS resolvers may not directly correspond to an actual threat. This is due to the passive role of open resolvers in DNS resolution. A malicious open resolver can perform its (malicious) actions only when it receives a domain name resolution request. If no user queries the malicious open resolver, the manipulated DNS record is essentially meaningless. At this point, we need to see how malicious open resolvers are actually queried by legitimate users. Moreover, it would be further important research topic to investigate how malicious open resolvers attract legitimate users.

On the other hand, we classified the malicious websites (IP addresses) using Cymon reports. By incorporating static/dynamic analysis of the contents, the websites can be further analyzed in terms of the contents they provide, and what contributes to the maliciousness flag. With the scope of this work being mainly the addressing space, we did not do any such comprehensive analysis on the hosted contents, asides from handpicked addresses to confirm the soundness of using Cymon. Designing a framework that detects the malicious response and analyze the contents from the malicious destination addresses is an orthogonal direction.

D. DNSSEC

DNSSEC is used by recursives to verify the authenticity of responses. While promising in reducing the threat of DNS manipulation attacks, its slow deployment [30], [31] still may leave those attacks possible, and our work is focused on how to measure DNS manipulation in the wild. Although DNSSEC is a way to prevent DNS manipulation, a significant number of attack attempts are still present, which is observed in the above analysis. Moreover, DNSSEC did not yet completely replace DNS, which leaves a threat to malicious behavior on DNS [30], [31]. Investigating how those possibilities change in light of DNSSEC deployment is an interesting future topic.

E. IPv6

In this work, we limited our scope of work to IPv4, as we were more interested in the comparative analysis of open resolver's behaviors, thus we did not conduct a measurement over IPv6 addresses. The historical scanning data in 2013 we had is only concerned with IPv4 addresses, and that was a decisive factor limiting us. As the use of IPv6 increases significantly, however, it is clear that extending this research is promising and needed. As there are IPv6 hitlists open to the public [32], it is possible to measure and analyze only in the limited areas of IPv6 addresses, which is a future work.

F. Open Resolver vs. IP Address

We note that the number of open resolvers can be inconsistent with the number of IP addresses we found. However, given that the landing mechanism to an open resolver is an address, and nothing else, we focus on enumerating those landing addresses (IP addresses of the open resolvers), although figuring out the number of the physical open resolvers that the addresses resolve to is an important but an orthogonal problem. We hypothesize that the number of physical open resolvers is perhaps smaller than that of the addresses, and figuring that out such in-depth analysis is left as future work.

G. Open Resolver vs. Non-Open Resolver

In this study, it is observed that a number of open resolvers incompliant in their operation. It would be another interesting work to compare the behaviors of the open resolvers to non-open resolvers, although it is challenging due to the closed nature of non-open resolvers, which requires a distributed global measurement tool (*e.g.,* RIPE Atlas). We leave this work of comparing the resolvers' behavior and exploring the potential reason of their differences as our future work.

VII. Related Work

Researchers have done a lot of work to understand open DNS resolvers and associated threats.

A. Internet-Wide Scanning

Durumeric *et al.* [17] proposed ZMap, a high-speed application to run Internet-wide scans capable of surveying the IPv4 address space within 1 hour on a single machine. In addition, Open Resolver Project [5] is a project that actively investigates DNS servers world-wide since March 2013 and regularly provides open resolver statistics on the web. One can browse information of open resolvers from March 2013 until January 2017. Moreover, Shadowserver [27] is an organization that conducts surveys related to Internet security, and they also conduct active measurements of open resolvers. Takano *et al.* [9] focused on DNS server software and their distribution and performed measurements.

B. DNS Measurement

A large body of work exists on analyzing DNS resolvers, however most of them focused only on a small subset of resolvers. Therefore, it is unclear if the observed results can be generalized to all resolvers around the globe. For instance, Sisson [15] analyzed open resolvers based on sampled scans that repeatedly queried the same set of resolvers, thus covering only a small fraction of all open resolvers. Furthermore, Jiang *et al.* [33] analyzed the caching behavior of resolvers. The authors identified an attack vector in DNS software that allows to extend the caching of domains even after they have been removed from the upper DNS hierarchy. Schomp *et al.* [34] randomly probed the IPv4 address space to enumerate DNS resolvers and distinguish between recursive DNS resolvers and DNS proxies. Furthermore, the authors closely analyzed the caching behavior of resolvers in more detail. Gao *et al.* [35] analyzed a large set of DNS query-response pairs collected from over 600 recursive DNS resolvers. They observed that although there is a great variation in the characteristics of the DNS traffic across networks, the behavior of resolvers within an organization is very similar. In addition, Scott *et al.* [36] analyzed DNS resolutions by probing the IPv4 address space for open resolvers. Top 10,000 Alexa domain names at the identified resolvers were queried to analyze the infrastructure of the Content Delivery Networks (CDNs). By deploying the automated clustering algorithms, they detected CDN deployments in their scanning results. Hao *et al.* [37] conducted a large-scale measurement to figure out the authoritative DNS deployment patterns of modern web services and their characteristics. In addition, Thomas and Mohaisen [38] conducted a measurement of the leakage of Tor's.onion in global DNS.

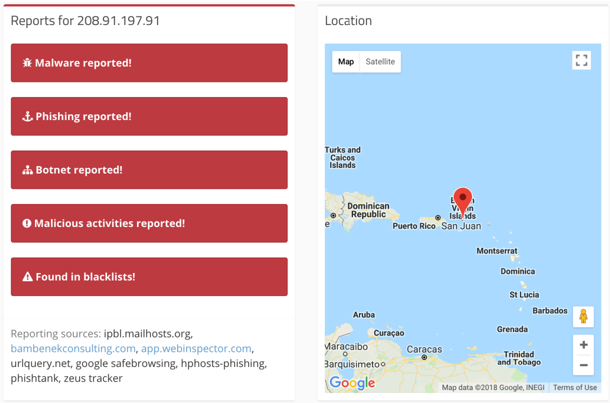
C. DNS Manipulation and Poisoning

As the role of DNS on the Internet becomes more and more important, research on DNS poisoning or manipulation has been actively conducted. Here, we introduce a couple of representative works about DNS manipulation. Antonakakis *et al.* [7] analyzed geographically diverse set of about 300,000 open recursive DNS servers and found that attackers generally point victims to rogue IP addresses. Kuhrer *et al.* [10] tried to shed light on the negative aspect of open resolvers, which can be abused by attackers. The authors measured the response authenticity of the resolvers from users' point of view and found that a large number of resolvers intentionally manipulate DNS resolutions. This work is similar to our study, but different in two points. First, we deal with more comprehensive behavioral aspects of open resolvers including how to fill the DNS header. Second, the previous work only focused on the manipulation for phishing, so they parsed and analyzed the HTTP file from the resolved address to identify the webpage for phishing purposes. In contrast, our work covers the wider scope of malicious activities including malware, phishing, botnet, *etc*. Schomp *et al.* [11] measured the vulnerability of the user-side DNS infrastructure to record injection threats and found that many open DNS resolvers, which are vulnerable to record injection attack, are being abused to attack shared DNS infrastructure. The recent work also highlighted that more than 92% of DNS resolution platforms are vulnerable to cache injection [39]. On the other hand, there have been lots of works to improve the consistency of DNS cache [40]–[42].

VIII. Conclusion

In this study, we conducted an up-to-date measurement of the distribution and behavior of open resolvers. Through an Internet-wide probing, we can see that about 3 million open resolvers still exist on the Internet and many of them operate in a way that deviates from the standard. From the result, we detected two threats posed by open resolvers.

First, the presence of millions of open resolver increases the threat of a DNS amplification DDoS attack. The adversary can exploit open resolvers as an amplifier by simply sending DNS 'ANY' queries to them, which results in the concentration of large DNS answers to the victim. Moreover, we also found evidence suggesting open resolvers' abnormal behaviors. The flag bits in the DNS response from open resolvers are often inappropriately marked. More than 69k open resolvers in 2018, for example, state that they are not recursion available (RA bit of 0) although they include the result of recursive resolution. More seriously, it is also shown that over 110k open resolvers provide the incorrect IP address as a DNS response, while more than 26k open resolvers return the IP addresses reported as malware, phishing, *etc*. Furthermore, through the deeper analysis of the packets collected at the authoritative server, we demonstrate the use of forwarders in the open resolver ecosystem and the possibility that incorrect or malicious responses are a by-product of the involvement of these forwarders. We plan to disclose the collected dataset and the modified source codes of Zmap to the public.


Reports for 208.91.197.91

Malware reported!

Phishing reported!

Botnet reported!

Malicious activities reported!

Found in blacklists!

Reporting sources: ipbl.mailhosts.org, bambenekconsulting.com, app.webinspector.com, urlquery.net, google safebrowsing, hphosts-phishing, phishtank, zeus tracker

Location

Fig. 6. The information in Cymon about the IP address of 208.91.197.91 that ranks in the third highest reference in 2018. Note the multiple reports about malware, botnet, phishing, etc. It can be assumed that the open resolvers which redirect the users to this address are exploited by the adversaries.

Appendix

A. DNS Responses With Empty `dns_question`

We briefly describe the analysis of 494 packets without `dns_question` field in 2018.

1) DNS Answer Presence: Among the 494 packets, 19 `R2` packets have the `dns_answer` field, which is about 3.8%. However, none of the 19 packets includes the correct answer. There are 14 packets containing a private network address in `dns_answer` (13 for 192.168.0.0/16, 1 for 10.0.0.0/8) and one with incorrect format (e.g., 0000). Moreover, 4 `R2` packets had addresses which could not be found in Whois.

2) RA Flag: 184 responses with `RA` bit of 1 were found. The 19 responses with incorrect IP address above had a `RA` bit of 1, and the other 165 packets did not include the resulted address even if they had an `RA` of 1. All the 303 packets with `RA` set to 0 did not contain `dns_answer`.

3) AA Flag: With `AA` flag, only two responses out of 494 had an `AA` bit of 1, and the rest did not. Only one of two responses contained `dns_answer`, although incorrect. The 19 `R2` packets with the answer field had the `AA` bit set to 0.

4) Response Code: Among the `R2` packets, 26 responses had an rcode of 0 (NoError), 1 response of 1 (FormErr), 301 responses of 2 (ServFail), 2 responses of 3 (NXDomain), and 163 responses of 5 (Refused). We can see that the failure and refusal are major reasons for the blank `dns_question`.

B. The Example of Malicious IP Address Reports

See Fig. 6.

References

[1] J. Park, A. Khormali, M. Mohaisen, and A. Mohaisen, "Where are you taking me? Behavioral analysis of open DNS resolvers," in *Proc. 49th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, Jun. 2019, pp. 493–504.

[2] *Public DNS–Google Developers*. Accessed: Aug. 17, 2021. [Online]. Available: https://developers.google.com/speed/public-dns/

[3] *Cloud Delivered Enterprise Security by Opendns*. Accessed: Aug. 17, 2021. [Online]. Available: https://www.opendns.com

[4] M. Kührer, T. Hupperich, C. Rossow, and T. Holz, "Exit from hell? Reducing the impact of ampli?cation DDoS attacks," in *Proc. USENIX Secur. Symp.*, 2014, pp. 111–125.

[5] *Open Resolver Project*. Accessed: Sep. 5, 2020. [Online]. Available: http://openresolverproject.org/

[6] D. Dagon, N. Provos, C. P. Lee, and W. Lee, "Corrupted DNS resolution paths: The rise of a malicious resolution authority," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2008, pp. 1–15. [Online]. Available: https://www.ndss-symposium.org/wp-content/uploads/2017/09/Corrupted-DNS-Resolution-Paths-The-Rise-of-a-Malicious-Resolution-Authority-paper-David-Dagon.pdf

[7] M. Antonakakis, D. Dagon, X. Luo, R. Perdisci, W. Lee, and J. Bellmor, "A centralized monitoring infrastructure for improving DNS security," in *Proc. Int. Workshop Recent Adv. Intrusion Detection (RAID)*, 2010, pp. 18–37.

[8] CloudFlare. (2013). *The DDoS That Knocked Spamhaus Offline (and How we Mitigated it)*. [Online]. Available: http://blog.cloudflare.com/the-ddos-that-knocked-spamhaus-offline-and-ho

[9] Y. Takano, R. Ando, T. Takahashi, S. Uda, and T. Inoue, "A measurement study of open resolvers and DNS server version," in *Proc. Internet Conf. (IC)*, 2013, pp. 23–32.

[10] M. Kührer, T. Hupperich, J. Bushart, C. Rossow, and T. Holz, "Going wild: Large-scale classification of open DNS resolvers," in *Proc. ACM Internet Meas. Conf. (IMC)*, 2015, pp. 355–368.

[11] K. Schomp, T. Callahan, M. Rabinovich, and M. Allman, "Assessing DNS vulnerability to record injection," in *Proc. Int. Conf. Passive Active Netw. Meas. (PAM)*, 2014, pp. 214–223.

[12] P. Pearce *et al.*, "Global measurement of DNS manipulation," in *Proc. USENIX Secur. Symp.*, 2017, pp. 307–323.

[13] N. Weaver, C. Kreibich, and V. Paxson, "Redirecting DNS for Ads and profit," in *Proc. USENIX Workshop Free Open Commun. Internet (FOCI)*, 2011, pp. 1–6.

[14] B. Ager, W. Mühlbauer, G. Smaragdakis, and S. Uhlig, "Comparing DNS resolvers in the wild," in *Proc. 10th Annu. Conf. Internet Meas. (IMC)*, 2010, pp. 15–21.

[15] G. Sisson. (2010). *DNS Survey: October 2010*. [Online]. Available: http://dns.measurement-factory.com/surveys/201010/dns_survey_2010.pdf

[16] J. Damas, M. Graff, and P. Vixie, *Extension Mechanisms for DNS (EDNS(0))*, IETF document RFC 6891, 2013.

[17] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast internet-wide scanning and its security applications," in *Proc. USENIX Secur. Symp.*, 2013, pp. 605–620.

[18] *GoDaddy*. Accessed: Aug. 17, 2021. [Online]. Available: https://www.godaddy.com/

[19] P. V. Mockapetris, *Domain Names–Implementation and Specification*, IETF document RFC 1035, 1987.

[20] P. V. Mockapetris, *Domain Names–Concepts and Facilities*, IETF document RFC 1034, 1987.

[21] D. Eastlake, *Domain Name System (DNS) IANA Considerations*, IETF document RFC 6895, 2013.

[22] *R. Tracker*. Accessed: Nov. 20, 2019. [Online]. Available: https://ransomwaretracker.abuse.ch/ip/208.91.197.91/

[23] *Cymon*. Accessed: Sep. 5, 2020. [Online]. Available: https://cymon.io/208.91.197.91

[24] *C. API*. Accessed: Sep. 5, 2020. [Online]. Available: http://docs.cymon.io/

[25] *IP2location*. Accessed: Aug. 17, 2021. [Online]. Available: https://lite.ip2location.com/

[26] *I. A. R. by Country (IP2location)*. Accessed: Aug. 17, 2021. [Online]. Available: https://lite.ip2location.com/ip-address-ranges-by-country

[27] *Shadowserver*. Accessed: Aug. 17, 2021. [Online]. Available: https://dnsscan.shadowserver.org/

[28] *Censys*. Accessed: Aug. 17, 2021. [Online]. Available: https://censys.io/data/

[29] *Rapid7*. Accessed: Aug. 17, 2021. [Online]. Available: https://opendata.rapid7.com/

[30] K. Fukuda, S. Sato, and T. Mitamura, "A technique for counting DNSSEC validators," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 80–84.

[31] Y. Yu, D. Wessels, M. Larson, and L. Zhang, "Check-repeat: A new method of measuring DNSSEC validating resolvers," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 381–386.

[32] O. Gasser *et al.*, "Clusters in the expanse: Understanding and unbiasing IPv6 hitlists," in *Proc. ACM Internet Meas. Conf. (IMC)*, 2018, pp. 364–378.

[33] J. Jiang, J. Liang, K. Li, J. Li, H. Duan, and J. Wu, "Ghost domain names: Revoked yet still resolvable," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2012, pp. 1–13. [Online]. Available: https://www.ndss-symposium.org/wp-content/uploads/2017/09/12_1.pdf

[34] K. Schomp, T. Callahan, M. Rabinovich, and M. Allman, "On measuring the client-side DNS infrastructure," in *Proc. Conf. Internet Meas. Conf.*, Oct. 2013, pp. 77–90.

[35] H. Gao *et al.*, "An empirical reexamination of global DNS behavior," in *Proc. ACM SIGCOMM Conf. SIGCOMM*, Aug. 2013, pp. 267–278.

[36] W. Scott, S. Berg, and A. Krishnamurth, "Satellite: Observations of the internet's star," Univ. Washington, Seattle, WA, USA, Tech. Rep. UW-CSE-2015-06-02, 2015.

[37] S. Hao, H. Wang, A. Stavrou, and E. Smirni, "On the DNS deployment of modern web services," in *Proc. IEEE 23rd Int. Conf. Netw. Protocols (ICNP)*, Nov. 2015, pp. 100–110.

[38] M. Thomas and A. Mohaisen, "Measuring the leakage of onion at the root: A measurement of Tor's .onion pseudo-TLD in the global domain name system," in *Proc. 13th Workshop Privacy Electron. Soc. (WPES)*, 2014, pp. 173–180.

[39] A. Klein, H. Shulman, and M. Waidner, "Internet-wide study of DNS cache injections," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, May 2017, pp. 1–9.

[40] X. Chen, H. Wang, S. Ren, and X. Zhang, "Maintaining strong cache consistency for the domain name system," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 8, pp. 1057–1071, Aug. 2007.

[41] S. Hao and H. Wang, "Exploring domain name based features on the effectiveness of DNS caching," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 47, no. 1, pp. 36–42, Jan. 2017.

[42] X. Chen, H. Wang, and S. Ren, "DNScup: Strong cache consistency protocol for DNS," in *Proc. 26th IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2006, p. 40.

Jeman Park received the B.Sc. degree from Korea University in 2016 and the Ph.D. degree in computer science from the University of Central Florida in 2020. Since 2020, he has been a Post-Doctoral Fellow at Georgia Institute of Technology, working at the Cyber Forensics Innovation Laboratory. His research interests include network/DNS security, cyber forensics, malware analysis, and web security.

Rhongho Jang (Member, IEEE) received the B.S., M.E., and Ph.D. degrees (Hons.) from Inha University, South Korea, in 2013, 2015, and 2020, respectively, and the Ph.D. degree from the Department of Computer Science, University of Central Florida, in 2020. He is currently an Assistant Professor with the Department of Computer Science, Wayne State University. His research interests lie in the area of software defined networks, network security, traffic measurement, and mobile security in general.

Manar Mohaisen received the master's degree in communications and signal processing from the University of Nice Sophia Antipolis, France, in 2005, and the Ph.D. degree in communications engineering from Inha University, Seoul, South Korea, in 2010. From 2001 to 2004, he worked as a Cell Planning Engineer at Palestinian Telecommunications Company. From 2010 to 2019, he was a full-time Lecturer and an Assistant Professor with the Department of EEC Engineering, Korea Tech, South Korea. Since August 2019, he has been working as an Assistant Professor with the Department of Computer Science, Northeastern Illinois University. His research interests include wireless communications with a focus on MIMO systems and social network analysis.

David Mohaisen (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from the University of Minnesota in 2012. He is currently an Associate Professor with the University of Central Florida, where he directs the Security and Analytics Lab (SEAL). Before joining UCF in 2017, he was an Assistant Professor at SUNY Buffalo (2015–2017) and a Senior Research Scientist at Verisign Labs (2012–2015). His research interests are in the areas of networked systems security, online privacy, and measurements. He is an Associate Editor of IEEE TRANSACTIONS ON MOBILE COMPUTING (TMC), IEEE TRANSACTIONS ON CLOUD COMPUTING (TCC), and IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS (TPDS). He has been a Senior Member of ACM since 2018. He is a Distinguished Speaker of ACM and a Distinguished Visitor of IEEE.