Analyzing DDoS on Root DNS Servers: What Makes Them Different?

Zoe Yejin Cho
University of Southern California
Los Angeles, USA
yejincho@usc.edu

***Abstract*—This paper surveys methodologies for mitigating Distributed Denial of Service (DDoS) attacks, with a specific focus on their applicability to root DNS servers. While DDoS attacks have been extensively studied, the unique traits of root servers often make general DDoS defense strategies ineffective or impractical. By comparing conventional DDoS mitigation techniques to those tailored for root servers, this survey identifies the limitations and gaps in existing approaches.**

The paper also analyzes the history of DDoS attacks on root servers, highlighting how their operational constraints and high availability requirements have influenced the evolution of defense strategies. Additionally, it explores the unintended consequences of some security enhancements, showing how certain measures intended to protect the DNS ecosystem may inadvertently exacerbate the impact of DDoS attacks. The survey concludes by proposing insights into optimizing link utilization and addressing the challenges posed by these critical vulnerabilities in root DNS infrastructure.

I. Introduction

The Domain Name System (DNS) serves as a cornerstone of the Internet, translating human-readable domain names into IP addresses and enabling seamless communication between users and networked services. At the apex of the DNS hierarchy are the root servers, a distributed network of 13 logical servers (labeled A to M), which form the first step in the DNS resolution process. These root servers operate under a highly resilient and distributed architecture, leveraging Anycast routing to ensure availability, redundancy, and low latency. Their critical role in the Internet ecosystem demands that they remain operational under all circumstances.

Despite their robust design, DNS root servers are not immune to threats, particularly Distributed Denial of Service (DDoS) attacks. DDoS attacks exploit the connectionless nature of DNS queries, flooding servers with malicious traffic to exhaust resources and disrupt service availability. While general DDoS mitigation techniques have been extensively studied, the unique operational constraints of root DNS servers—such as their stringent availability requirements, lack of discrimination policies, and resistance to structural changes—make conventional approaches either ineffective or impractical. These servers must balance their role as critical infrastructure with the challenge of mitigating high-volume attacks without disrupting legitimate traffic.

The history of DDoS attacks on root DNS servers highlights the evolving nature of these threats. Events such as the 2015 root DNS attack and subsequent traffic surges in later years demonstrate that while root servers have successfully maintained their resilience, the scale and sophistication of attacks are increasing. Analysis of past incidents reveals the diverse attack vectors employed, ranging from UDP flooding and DNS amplification to misconfigurations and unintentional query surges. Additionally, new technologies intended to enhance DNS security and privacy, such as DNSSEC and Query Name Minimization (QMIN), have inadvertently introduced complexities that exacerbate the risk of DDoS attacks.

This paper surveys existing methodologies for mitigating DDoS attacks, with a specific focus on their applicability to root DNS servers. It compares conventional defense mechanisms—such as rate limiting, Anycast, and layered defenses—to innovative approaches like BGP Flowspec, which enables granular traffic filtering closer to the source. The survey also examines the unintended consequences of emerging DNS technologies and explores the challenges of distinguishing malicious traffic from legitimate yet problematic query patterns.

By analyzing historical DDoS events, the operational nuances of root servers, and current defense strategies, this paper aims to provide insights into optimizing mitigation approaches while maintaining the reliability and resilience of the DNS root server system. The findings underscore the importance of collaborative efforts between root server operators, Internet Service Providers (ISPs), and the broader DNS community to address these critical challenges.

II. Background

A. DNS Iterative Resolution System

The Domain Name System (DNS) employs an iterative resolution process to efficiently map human-readable domain names to IP addresses. This process involves multiple steps, starting from the recursive resolver and proceeding through the hierarchy of DNS servers, including root servers, top-level domain (TLD) servers, and authoritative servers.

In an iterative system, the responsibility for resolving a domain name is distributed across different DNS servers. The process begins when a user initiates a query through a stub resolver, which forwards it to a recursive resolver, often managed by an Internet Service Provider (ISP). The recursive resolver communicates with other DNS servers to resolve the query step-by-step. At each step, the server provides a referral

to the next level in the DNS hierarchy rather than the final answer.

Root servers are the starting point of the DNS hierarchy. When a recursive resolver needs information about a domain (e.g., `www.example.com`) and lacks a cached response, it queries a root server. The root server does not have the complete answer but provides a referral to the relevant TLD server (e.g., `.com`). It is important to note that root servers are not queried directly by end-users; instead, queries are sent by recursive resolvers typically managed by ISPs or large public DNS providers. This structure minimizes unnecessary traffic to root servers and optimizes query efficiency.

The iterative resolution process involves several steps. First, the user's stub resolver sends the query (e.g., `www.example.com`) to a recursive resolver. The recursive resolver then queries a root server for the domain, and the root server replies with a referral to the appropriate TLD server. Next, the recursive resolver contacts the TLD server (e.g., `.com` server) to get a referral to the authoritative server for `example.com`. Finally, the recursive resolver queries the authoritative server, which provides the IP address for `www.example.com`.

Recursive resolvers are generally managed by ISPs or public DNS providers to handle these iterative queries on behalf of end-users. This arrangement reduces the load on root servers and ensures faster responses for users by leveraging cached results for frequently queried domains. Without this hierarchical system, root servers would face an overwhelming number of queries, compromising DNS reliability.

The iterative resolution system offers several advantages. It enhances efficiency by caching responses at the recursive resolver level, reducing repeated queries to root and authoritative servers. Its hierarchical design ensures scalability by distributing the load across different DNS layers. Moreover, the system's resilience minimizes dependency on a single server, enhancing fault tolerance. This approach ensures that DNS can scale to support billions of queries daily, while ISPs and DNS providers play a important role in balancing efficiency and reliability.

B. Role of Anycast in DNS

Anycast is a technique that enhances the scalability, resilience, and performance of Internet services by allowing multiple servers across the globe to share the same IP address. It is reliant on BGP (Border Gateway Protocol) to dynamically route traffic to the nearest server. This mechanism is fundamental in the functioning of widely used services like DNS, particularly root servers, as it enables traffic to be efficiently distributed across multiple locations. Anycast also ensures the safety and reliability of these services, helping them remain robust and resistant to failures or attacks by diversifying traffic destinations and preventing overloading on a single server [16].

Anycast is a network routing method where multiple servers across various geographic locations share the same IP address. User requests to this address are automatically routed to the nearest or most optimal server based on the BGP table, improving response times and distributing traffic loads for better reliability.

For B-root, Anycast provides good benefits. It allows deployment of multiple servers under the same IP address across different regions, enabling user requests to be routed to the nearest server. This reduces latency and enhances response times. Additionally, Anycast balances traffic across servers, mitigating risks of congestion or overload. It also improves resilience by eliminating a single point of failure—if one server goes down, others can handle requests, ensuring high availability and stability. This strengthens B-root's role in maintaining a reliable global DNS system. You can see the figure 2 to see the correlation between geolocation of IP blocks and anycast catchment.

Also, Anycast is widely adopted by Content Distribution Networks (CDNs). Examples are Microsoft/Azure [17] and Verizon/Edgecast [18].

III. DNS Server Specifics

DNS servers play a important role in internet infrastructure but are not immune to exploitation. This section discusses their potential use as attack vectors, limitations in their architecture, and specific attacks applicable or not applicable to root DNS servers.

A. Inherent Challenges of DNS Servers

While DNS servers are vital, certain design principles and configurations impose limitations:

- **Avoidance of Single Points of Failure:** Devices or architectures that could act as a single point of failure are unsuitable for DNS systems. For instance, while Software-Defined Networking (SDN) offers centralized control, the Controller can be the Single Points of Failure, thus not suitable for Root DNS systems.
- **Autonomy in Root Servers:** The distributed organization of the 13 root servers (labeled A to M) ensures resilience and reliability, with autonomy being a cornerstone of their operation. Each root server is independently managed by a different organization or entity, guaranteeing that no single point of control exists. However, this independence also means that the root servers operate without a formal framework for collaboration, limiting their ability to coordinate responses or share resources effectively in real-time scenarios. While this lack of centralized management enhances decentralization and fault tolerance, it presents challenges in implementing uniform changes or addressing large-scale network issues. The balance between autonomy and cooperation is crucial to maintaining a robust DNS infrastructure [15].
- **Availability of Root Servers:** Root servers are designed to prioritize reliability and availability over all else, aiming to ensure no DNS requests are dropped. Unlike many other online services, root servers cannot afford downtime or service interruptions, as DNS resolution is

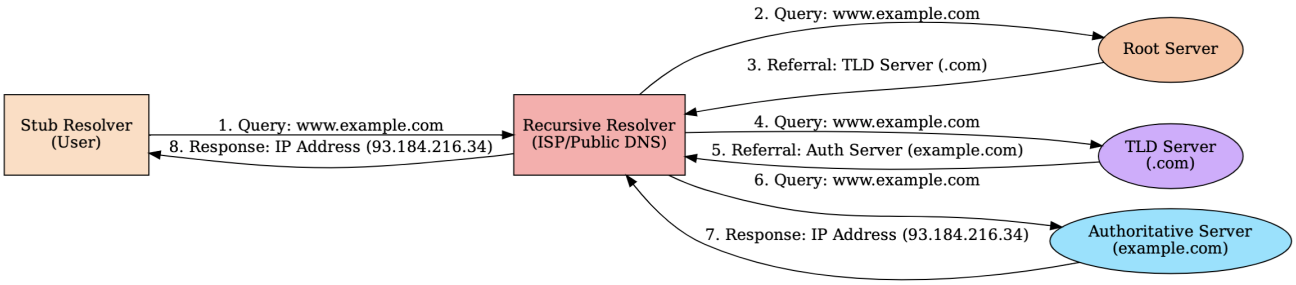

Fig. 1: DNS Recursive Resolution Process.

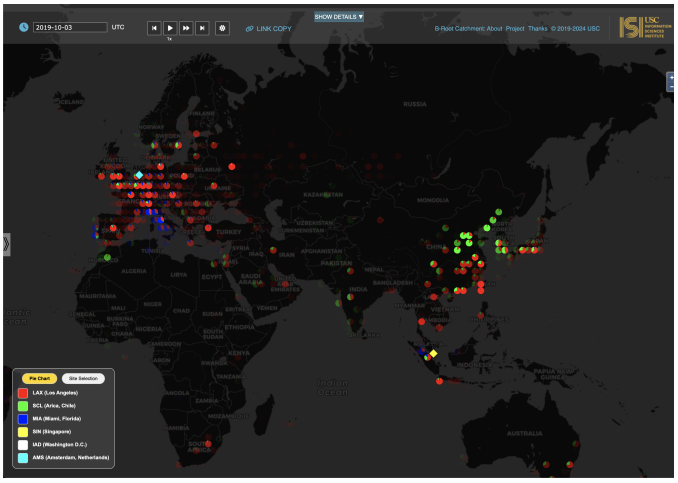

Fig. 2: Anycast visualization for B-root. This figure demonstrates the global distribution and routing efficiency of Anycast for B-root DNS servers, showcasing how traffic is directed to the nearest server based on geographic proximity. (Created by the author)

important for the functioning of the internet. This stringent requirement makes them fundamentally different from other systems that might allow unavailability.

- **No Discrimination Policy:** Root DNS servers operate under a strict policy of neutrality and fairness, meaning they cannot discriminate among users based on their origins or access patterns unless there is concrete evidence of an attack. The most common users of root servers are ISPs and large-scale DNS resolvers, but this does not justify prioritizing these users over others. For instance, a first-time user querying a DNS resolver should experience the same performance as any other user. This policy ensures consistent and equitable access, though it complicates mitigation strategies for potential Distributed Denial of Service (DDoS) attacks [1].
- **Resistance to Structural Changes:** The DNS protocol and root server infrastructure are designed for long-term stability, meaning changes to their structure or functionality must be approached with extreme caution. For example, DNS operates at Layer 3, which limits the ability to adopt defenses like DNS over HTTPS (DoH) or DNS over TLS (DoT) without substantial protocol adjustments. Backward compatibility is a important requirement, ensuring that legacy systems continue to function seamlessly. Any proposed change must demonstrate clear and significant advantages before it can be implemented across the system [12], [15]. However, some root servers may try new systems. Recently, B-Root has begun support for DNS-over-TLS (DoT) on port 853 on an experimental basis. [24]

B. DNS Servers Must Prevent Exploitation as Reflection Servers

DNS servers can be exploited as reflectors in Distributed Reflection Denial of Service (DRDoS) attacks. A notable example is the **DNS Amplification Attack**, a type of reflection-based DDoS attack where an attacker sends small DNS queries with a spoofed IP address, resulting in large responses that overwhelm the target. This amplification effect is comparable to a “Smurf Attack” in concept but utilizes DNS queries instead.

C. Attacks Not Feasible Against Root DNS Servers

Root DNS servers are designed with mechanisms to resist common types of network attacks. The following attacks are impossible:

- **TCP SYN Flooding:** Root DNS servers primarily use UDP for communication and only establish TCP connections under specific circumstances, limiting the impact of SYN Flood attacks.
- **HTTP Flooding:** Root servers respond exclusively to DNS queries, and will drop HTTP-based attacks.
- **ICMP Flooding (Ping Flooding):** Most root servers restrict or block ICMP requests, neutralizing this attack vector.
- **ANY Query Exploitation:** Root servers return minimal information or empty responses to ‘ANY‘ queries. This behavior aligns with RFC 8482, which specifies minimizing response sizes to protect server resources.

TABLE I: Effectiveness of Different Attack Types on Root DNS Servers

Attack Type	Effectiveness	Reasons
TCP SYN Flooding	Not Effective	Limited use of TCP connections by root DNS servers.
HTTP Flooding	Not Effective	Root servers respond only to DNS queries, not HTTP requests.
ICMP Flooding	Not Effective	ICMP requests are restricted or blocked.
ANY Query Exploitation	Not Effective	Minimal responses to ANY queries as per RFC 8482.
UDP Flooding	Effective	UDP's connectionless nature allows high-volume traffic to overwhelm servers.
DNS Amplification	Effective	Leveraging recursive resolvers to amplify queries.

IV. Attacks Feasible Against Root DNS Servers

Despite their robust design, root DNS servers remain susceptible to certain attacks:

- **UDP Flooding:** As root servers primarily communicate via UDP—a connectionless protocol—they can be overwhelmed by large volumes of UDP packets. UDP's lack of connection establishment makes it particularly advantageous for attackers aiming to send high-velocity traffic to overload the server.
- **DNS Amplification:** Exploiting the UDP-based nature of DNS, attackers can leverage recursive resolvers to amplify queries and generate significant traffic volumes, overwhelming root servers.

Root DNS servers are configured to prioritize resilience and security. By adhering to standards such as RFC 8482, limiting TCP connections, and mitigating ICMP traffic, they reduce vulnerabilities while maintaining DNS functionality.

V. Analyzing Previous DDoS Attacks on Root DNS Servers

While root DNS servers have occasionally been the target of DDoS attacks, comprehensive traffic analyses suggest that many *DDoS-like* events are attributable to misconfigurations or unintended traffic surges. However, some incidents demonstrate clear malicious intent, as evidenced by their patterns and impact.

A. Previous DDoS Events: Public Information

- **2015 Event:** A Distributed Denial of Service (DDoS) attack targeted the root DNS servers, reaching a peak traffic volume of 50 Gbps. This event demonstrated the ability of attackers to generate massive traffic loads aimed at overwhelming internet infrastructure. Despite the scale of the attack, the root server system remained operational due to its distributed and autonomous nature, which helped mitigate the impact. The event highlighted the growing sophistication and scale of DDoS attacks targeting DNS infrastructure, as well as the importance of robust defense mechanisms [3].
- **2016-06-25 Event:** On June 25, 2016, a significant traffic surge was observed at the root name servers, with each root name server letter (A through M) handling approximately 10 million packets per second. This equated to around 17 Gbps of traffic per root server letter. [23]

B. Previous DDoS Events: Private Information

Analysis of anomalies has been documented for B-root in a private repository on the ANT wiki, which is not publicly accessible [5]. The documented analysis is based on data derived from DITL (Day-in-the-Life of the Internet) [6], a periodic data collection of root server activity. This data has provided insights into patterns of abnormal query rates, and potential attacks targeting the B-root server.

Table II provides a summary of notable DDoS attack events targeting B-root, including descriptions of attack patterns, observed query names, and offered load during each event. These incidents illustrate a wide range of attack methodologies, from randomized source IPs and query names to more sophisticated TCP SYN flood attacks and DNS reflection techniques. For instance, the attack on 2015-11-30 featured randomized IPs querying common domains such as `www.336901.com`, while the event on 2020-02-14 involved reflection attacks leveraging DNSSEC-signed responses. The diversity of these attack vectors underscores the importance of adaptive and robust mitigation strategies in safeguarding root server operations.

C. Challenges

One of the significant challenges in managing DNS servers, especially root servers, is the difficulty of distinguishing between malicious activity and unintentional errors. Root servers handle an immense volume of queries, making it challenging to identify abnormal traffic patterns without inadvertently flagging legitimate activity as an attack. Misconfigurations in DNS resolvers, such as poorly implemented retry mechanisms or software bugs, often lead to excessive queries that mimic the behavior of a Distributed Denial of Service (DDoS) attack. Similarly, IoT devices, which frequently lack proper DNS configuration, can unintentionally generate high traffic volumes that resemble attack patterns. To complicate matters further, attackers often spoof legitimate IP addresses during DNS-based attacks, making it difficult to trace traffic back to its true origin. The global diversity of queries to root servers also adds complexity, as legitimate international traffic can closely resemble geographically distributed attack traffic, such as that from botnets. Additionally, root servers operate under a strict non-discrimination policy, which means they cannot block or filter traffic without concrete evidence of malicious intent. Blocking legitimate queries could disrupt DNS resolution for many users, undermining the stability and reliability of the internet.

TABLE II: Attack Days and Common Query Names

Attack Day	Attack Description	Duration	Observed Offered Load
2015-11-30	Randomized IPs, common query names were `www.336901.com` and `www.366901.com` with Response code 0 in replies.	06:50 UTC to 09:19 UTC	0.30 to 0.38 Mqps
2015-12-01	Randomized IPs with `www.916yy.com` query name which had Response code 0 in replies.	05:10 UTC to 06:13 UTC	0.30 to 0.38 Mqps
2016-06-25	Randomized IPs, TCP SYN flood attack.	22:18 UTC to 23:59 UTC	0.084 Mqps
2017-02-21	Not randomized IPs, common query names were `RANDOM.jiang.comSPACE`, `RANDOM.phone.tianxintv.cnSPACE`, and `RANDOMclgc88.comSPACE`, with NXDomain replies.	06:40 UTC to 08:37 UTC (with prior and post spikes)	0.08 to 0.12 Mqps
2017-03-06	Not randomized IPs, common query names were `RANDOM.qycl520.comSPACE` and `RANDOM.cailing168.comSPACESPACE`, with NXDomain replies.	04:43 UTC to 10:14 UTC (with prior and post spikes)	0.08 to 0.10 Mqps
2017-04-25	Not randomized IPs, common query name was `RANDOM.plaza.game981.comSPACE`, with NXDomain replies.	09:54 UTC to 12:49 UTC	0.08 to 0.10 Mqps
2019-09-07	No fixed query name (all were 554-byte requests).	06:45:19 UTC to 06:46:53 UTC	0.80 to 1.0 Mqps (LAX site), 0.10 to 0.12 Mqps (ARI site), 1.0 to 1.2 Mqps (MIA site)
2020-02-13	Attack from `61.220/16` network (e.g., `61.220.3/24`, `61.220.7/24`, and `61.220.11/24`), DNS over UDP, some TCP SYN packets. Query name: `RANDOM.8.8.8.8`, mostly NXDomain replies.	08:05 UTC to 08:08 UTC	0.3 Mqps (SIN site)
2020-02-14	Mostly at LAX. Reflection attack directed at root server. Fragmented packets, CLDAP packets, DNS response packets (with DNSSEC signatures). Worth IP/TTL/hopcount filtering studies.	23:17:47 UTC to 23:19:00 UTC	0.12 Mqps (LAX site)
2020-02-17	Randomized IPs, DNS over UDP and TCP queries with `ANY` type and `<Root>` as query name. Multiple TCP SYN packets with several connections per source.	20:18 UTC (SIN site), repeated attacks at 21:31, 21:49, and 21:58 UTC.	Not tested
2020-10-24	Mostly at SIN site. Real TCP SYN flood attack with spoofed sources.	02:55 UTC to 03:00 UTC (SIN site)	2.3M pkts/s
2021-05-28	All sites - randomized sources, IP fragmented queries (large packets). Server fixed 60-byte replies. DNSMon reported no loss for IPv4, 33% loss for IPv6. Query name: `pizzaseo.com`.	02:35 UTC (3–5 minutes)	60 Gb/s (across all sites)

D. Traffic Analysis and Insights

Analysis of DNS traffic at B-Root by Ginesin et al. [19] highlights the evolving composition of DNS queries and the challenges posed by high volumes of malformed traffic, which are not necessarily malicious but can still stress infrastructure. Using a decade of data from CAIDA's Day in the Life of the Internet (DITL [6]) project, the study found that malformed queries increased from 39.57% in 2013 to 67.91% in 2022, as illustrated in Figure 3.

A substantial proportion of malformed queries originate from large-scale platforms like Amazon Web Services (AWS) and Microsoft Azure. In 2022, AWS accounted for 14% and Microsoft Azure for 3% of all queries to B-Root, with malformed traffic dominating in both cases (Figure 4). Notably, the majority of this traffic does not resemble high-intensity bursts typical of Distributed Denial of Service (DDoS) attacks. Instead, it reflects persistent, misconfigured behavior, such as queries with invalid top-level domains (TLDs) or repeated requests from improperly configured systems. This steady stream of malformed traffic, while not immediately disruptive, can cumulatively place a significant burden on DNS infrastructure.

Chromium-based browsers, particularly through their Omnibox feature, further illustrate this issue. Omnibox generates randomized DNS queries to test connectivity, inadvertently flooding root servers with queries that are indistinguishable from malformed traffic caused by misconfigurations. At its peak in 2020, Chromium was responsible for nearly 50% of all queries to B-Root (Figure 5). While such traffic is not malicious, its volume highlights how benign design decisions can result in unintentional stress on the DNS system.

Rather than indicating malicious intent, the observed traffic reflects the interplay of new technologies, increasing reliance on cloud platforms, and persistent misconfigurations. These findings suggest the need to shift focus from solely detecting attacks to proactively managing the impact of non-malicious but problematic traffic. Advanced monitoring tools and better collaboration with large-scale platforms like AWS and Chromium are essential to reduce the strain on root DNS

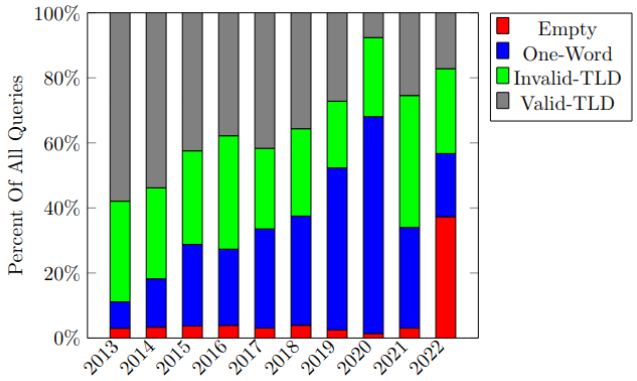

Fig. 3: Increase in malformed DNS queries at B-Root from 2013 to 2022. The percentage of malformed queries has steadily risen, highlighting persistent misconfigurations and evolving traffic patterns. Source: Ginesin and Mirkovic [19].

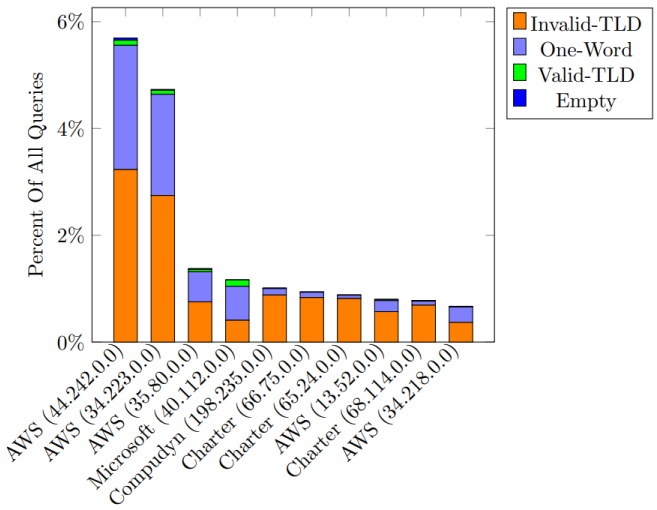

Fig. 4: Breakdown of top query senders to B-Root in 2022. Amazon Web Services (AWS) and Microsoft Azure account for a significant portion of queries, most of which are malformed. Source: Ginesin and Mirkovic [19].

infrastructure while maintaining robust service.

VI. Defense Mechanisms

A. Traffic Shaping

Traffic shaping, particularly through rate limiting, is one of the most commonly employed methods to mitigate DDoS attacks. By enforcing predefined thresholds on traffic volume, rate limiting effectively prevents server overload. Specific implementations include source filtering, which blocks traffic from identified malicious sources, and geo-fencing or geo-blocking, which restricts traffic based on geographic regions. While these methods are straightforward and effective, they are not without drawbacks. Legitimate requests may inadvertently be blocked, and the configuration process can be complex, potentially increasing latency during implementation.

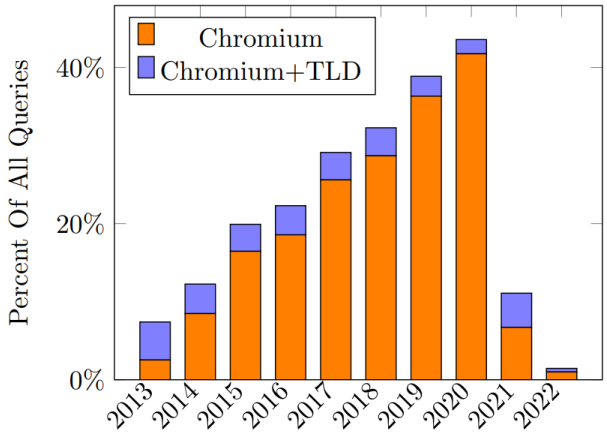

Fig. 5: Growth of Chromium-initiated DNS queries at B-Root from 2013 to 2022. The Omnibox feature led to a significant surge in malformed queries, peaking at nearly 50% of all queries in 2020. Data source: Ginesin and Mirkovic [19].

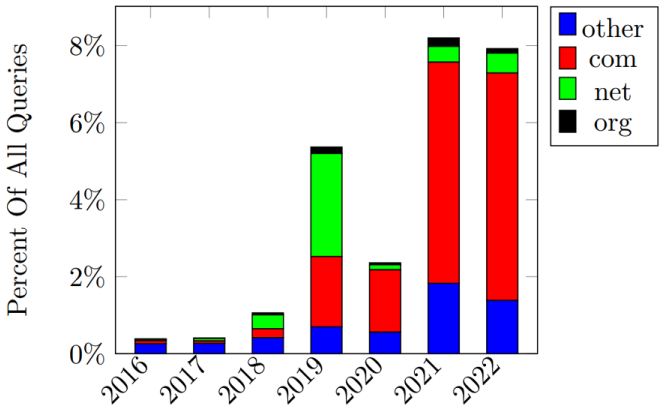

Fig. 6: Increase in Query Name Minimization (QMIN) queries since the introduction of RFC 7816 in 2016. QMIN adoption has steadily grown, reflecting the evolving nature of DNS traffic. Data source: Ginesin and Mirkovic [19], based on CAIDA's DITL project [6].

B. Anycast

Anycast, a routing strategy that distributes DNS queries across multiple geographically dispersed servers, is another widely used defense mechanism. This approach enhances scalability by spreading traffic load, thereby reducing the latency experienced by end users. However, deploying and maintaining an Anycast network is resource-intensive, and attackers can still exploit the network by targeting its weakest points, making this solution less foolproof than it may appear.

C. Layered Defenses

Layered defenses, proposed by Rizvi et al. [13], offer a robust mechanism specifically designed for root DNS servers. By combining multiple mitigation techniques, this approach

ensures dynamic adaptation to diverse attack types without imposing significant operational overhead. Layered defenses are particularly advantageous in maintaining continuity of DNS services under adverse conditions. However, their effectiveness is contingent upon accurate attack detection, and the complexity of implementing and maintaining such systems poses additional challenges.

D. Change the Architecture itself, removing attack surface

A more innovative solution involves fundamentally altering the DNS architecture to minimize the attack surface. Allman [14] suggested enabling recursive resolvers to download static root zone files rather than querying root servers repeatedly. This architectural change significantly reduces the query volume directed at root servers, thereby enhancing their resilience to DDoS attacks. Distributing root zone files also minimizes the load on root servers. However, this approach introduces operational complexities, such as ensuring file consistency and the timely distribution of updates.

E. Deep Learning for DDoS Detection

In recent years, deep learning has emerged as a promising solution for detecting DDoS attacks. For instance, machine learning models integrated into Software-Defined Networking (SDN), as demonstrated in [26], have shown high accuracy in identifying complex attack patterns. These models are adaptive to evolving threats, making them highly effective in theory. Nonetheless, their practical deployment is hampered by significant computational costs and operational challenges. Real-time detection requires robust and scalable algorithms, which may introduce delays, particularly at the scale required for root DNS servers.

Defending DNS servers against DDoS attacks requires a combination of traditional methods, like rate limiting and Anycast, and advanced techniques, such as layered defenses and deep learning. However, practical concerns like cost, scalability, and operational overhead must be carefully considered when choosing a defense strategy.

F. My Proposal: Leveraging Flowspec

BGP Flowspec provides a precise and innovative approach to mitigating Distributed Denial of Service (DDoS) attacks, particularly when combined with the distributed nature of Anycast routing. Flowspec allows for highly granular filtering of traffic based on characteristics like protocol, port, and payload, enabling malicious traffic to be dropped closer to its source while preserving legitimate flows. Unlike traditional methods such as Remote Triggered Black Hole (RTBH), which block entire destination IPs and risk collateral damage, Flowspec minimizes disruption by targeting specific traffic patterns [10].

By collaborating with Internet Service Providers (ISPs) to enforce these rules at upstream routers, malicious traffic originating from botnets and compromised devices can be blocked before it even enters important network links. Such collaboration is practical given existing adoption of Flowspec by major ISPs like AT&T [22], which already peers with B-Root, allowing for seamless deployment of filtering rules through established peering agreements.

ISPs may have a strong incentive to participate, as a significant portion of DDoS traffic originates from compromised IoT devices within their customers' networks. By mitigating attacks at the source, ISPs not only protect critical infrastructure but also enhance their reputation as responsible providers. Customers are increasingly aware of the risks posed by insecure IoT devices and would favor ISPs that actively address these challenges.

Such collaboration is practical given the existing adoption of Flowspec by major ISPs like AT&T [22], which already peers with B-Root through Los Nettos. This allows for seamless deployment of filtering rules through established peering agreements, preventing malicious traffic from botnets and compromised devices from consuming bandwidth or reaching critical network links.

G. Overall Comparison of DDoS Defense Methods

Table III compares several DDoS defense strategies based on key factors, including operational overhead, scalability, and deployment for root DNS servers.

VII. Discussion: Unintended Side Effect on DDoS Mitigation

While various advancements have been introduced to enhance DNS functionality and security, some have inadvertently exacerbated the challenges posed by DDoS attacks. This section explores several such technologies and their implications for root DNS servers.

A. DNSSEC: Enhancing Security but Increasing DDoS Risks

DNSSEC (DNS Security Extensions) was introduced to address vulnerabilities such as spoofing attacks, as demonstrated by incidents like the discovery of an unauthorized root server instance in China [27], which underscored the need for secure DNS resolution. However, while DNSSEC enhances data integrity, its design can inadvertently exacerbate DDoS attacks. DNSSEC responses are significantly larger than standard DNS responses, increasing the amplification factor when leveraged in reflection attacks. Furthermore, attackers can exploit DNSSEC-enabled servers to generate substantial volumes of traffic directed at a target. A comprehensive study by Van Rijswijk-Deij et al. [12] quantified this amplification effect, raising concerns about the unintended consequences of DNSSEC implementation.

B. QNAME Minimization: Privacy at the Cost of Detection Accuracy

QNAME Minimization reduces the amount of query information sent to DNS servers to enhance user privacy. However, this approach can hinder DDoS detection and mitigation by limiting the availability of full QNAME data. Security systems that rely on query patterns for anomaly detection or traffic filtering lose critical context, reducing their accuracy and effectiveness. While DNSSEC, DoH, DoT, and QNAME Minimization aim to improve security and privacy, they introduce

TABLE III: Comparison of DDoS Defense Strategies

Method	Description	Overhead	Scalability	Effectiveness	Ease of Deployment
Layered Defenses (Rizvi et al.)	Combines multiple techniques for mitigation; dynamically adapts to attack types.	Low	High	High	Medium
Machine Learning (e.g., [26])	Uses supervised learning to detect patterns in network traffic indicative of DDoS.	High	Medium	High	Low
Anomaly-Based Detection	Monitors traffic for unusual patterns, triggering alerts.	Medium	High	Medium	Medium
Rate Limiting	Limits traffic to predefined thresholds to prevent server overload.	Low	Medium	Low	High
Anycast Routing	Distributes traffic across multiple geographically distributed servers.	Low	High	High	Medium
DNS Traffic Filtering	Filters queries based on predefined rules to block malicious traffic.	Medium	High	Medium	Medium
BGP Flowspec	Enforces granular filtering rules to block malicious traffic closer to the source.	Medium	High	High	Low
Static Zone File Distribution [14]	Reduces query volume to root servers by enabling recursive resolvers to download static files.	Low	Medium	Medium	Low

complexities that can amplify the impact of DDoS attacks. Innovative ideas like reducing query dependency on root servers show promise but must be balanced with operational feasibility. These insights emphasize the importance of carefully assessing the trade-offs between security enhancements and their unintended consequences.

The adoption of new DNS protocols also introduces traffic patterns that differ from traditional behavior. For example, Query Name Minimization (QMIN) and priming queries, as defined in RFC 7816 and RFC 8109, respectively, have grown steadily in recent years. QMIN queries, which reduce the amount of information sent upstream, accounted for a significant share of traffic after their introduction in 2016 (Figure 6). However, priming queries, designed to initialize DNS resolver caches, saw a sudden surge in 2022, jumping to 35% of all queries. These changes in traffic composition can complicate the interpretation of anomalous behavior, as legitimate patterns evolve alongside misconfigurations.

VIII. Future Research Directions

Flowspec's effectiveness increases when combined with the distributed nature of Anycast routing. Research is needed to optimize the interaction between Flowspec rule enforcement and Anycast traffic distribution. We should investigating how Flowspec rules can be enforced closest to the source of malicious traffic within Anycast networks to reduce latency and minimize collateral impact. Also, for global rule coordination we should develop mechanisms to coordinate Flowspec rule enforcement across geographically distributed Anycast instances, ensuring consistent filtering without excessive overhead.

IX. Related Works

A. Management and Monitoring Tools

Effective management and monitoring tools are essential for detecting DDoS on DNS infrastructure:

- **DNSMon:** A comprehensive tool for monitoring DNS performance and availability, covering not only root servers but the broader DNS ecosystem [7].
- **Anycast B-root:** A visualization and analysis tool developed to monitor B-root traffic patterns and enhance operational insights. This tool was developed by the author (me), leveraging advanced techniques for traffic monitoring and analysis, and is accessible at anycast-b-root.ant.isi.edu.
- **Visual Analytics for Root DNS Data:** Advanced visualization techniques, as discussed by Krokos et al., provide intuitive ways to analyze root DNS data and identify anomalies in traffic patterns [4].
- **External Monitoring Systems:** As mentioned in Root Server Operators report([25]), some large events such as 2015 event can be observed in External Monitoring Systems.

X. Limitations: Why Root Servers Are Not Tempting Targets

Root DNS servers are less appealing as targets for DDoS attacks due to their robust design and specific operational characteristics:

- **Minimal Service Disruption:** DNS caching mechanisms significantly reduce the impact of attacks on root servers. Recursive resolvers typically cache responses from root servers for a Time-To-Live (TTL) period of about 20 minutes, meaning end-users can continue accessing cached data even during an attack. Additionally, Top-Level Domain (TLD) server addresses, which root servers primarily return, rarely change. This stability further limits the immediate impact of an attack on the broader DNS resolution process.
- **Low Amplification Coefficients:** Unlike open resolvers, root DNS servers are not optimized for providing large responses to queries. Their amplification factor is relatively low compared to other DNS infrastructure components, such as recursive resolvers. This makes root servers less attractive for use in DNS Amplification Attacks, as attackers seek higher amplification ratios to maximize the impact of their attacks.

- **Redundant Infrastructure:** The root DNS server system is globally redundant, consisting of 13 logical root servers (labeled A to M) that operate across hundreds of physical locations worldwide through Anycast routing. This distributed infrastructure ensures resilience against localized failures, as traffic can be redirected to alternative nodes if certain servers are overwhelmed. The independence of root server operators also minimizes the risk of systemic failure.
- **Hackers' Motivations and Target Analysis:** While root servers play a critical role in the DNS hierarchy, they are not the most attractive targets for attackers. Recursive and open resolvers are often preferred because they are more vulnerable to exploitation, such as through higher amplification factors, and their disruption directly impacts end-user experiences. Additionally, modern attackers tend to focus on targets with direct monetary or strategic value, such as web applications, e-commerce platforms, or cloud services, rather than infrastructure components like root servers.

XI. Conclusion

The DNS root server system represents a cornerstone of the internet's resilience and reliability, designed with principles of redundancy, autonomy, and distributed operations. Defending this infrastructure against DDoS attacks is important, but as many survey papers emphasize, introducing changes to this system must be approached cautiously. Even minor adjustments to the root server architecture can lead to significant challenges, including operational complexity and unintended consequences.

Acknowledgments

The author would like to thank ASM Rizvi for his foundational work on ANT wiki, Professor Heidemann for allowing to use ANT dataset for CSCI530 class research paper and Professor Neuman for feedback on choosing references.

References

[1] Mockapetris, P. (1987). "Domain names - concepts and facilities." *RFC 1034*. Internet Engineering Task Force.

[2] Root Server Operators. (2024). "The Root Server System Advisory Committee (RSSAC) Documents." https://www.icann.org/groups/rssac/documents

[3] Moura, G. C. M., et al. (2016). "Anycast vs. DDoS: Evaluating the November 2015 Root DNS Event." *ACM Internet Measurement Conference (IMC)*.

[4] E. Krokos, A. Rowden, K. Whitley, and A. Varshney, "Visual analytics for root DNS data," in *Proceedings of the 2018 IEEE Symposium on Visualization for Cyber Security (VizSec)*, 2018, pp. 1–8. [Online]. Available: https://www.computer.org/csdl/proceedings-article/vizsec/2018/08709205/19ZL29SopcQ

[5] "B-root Anomaly Analysis," ANT Wiki. Private Documentation, Information Sciences Institute (ISI).

[6] "DITL: Day in the Life of the Internet," Information Sciences Institute (ISI), https://ant.isi.edu/software/ditl/index.html

[7] "DNSMon: Monitoring DNS Performance," RIPE NCC. Available at https://dnsmon.ripe.net

[8] "Anycast B-root," Information Sciences Institute (ISI), https://anycast-b-root.ant.isi.edu

[9] P. Marques, et al. (2009). "Dissemination of Flow Specification Rules." *RFC 5575*, Internet Engineering Task Force.

[10] P. Marques, et al. (2009). "Dissemination of Flow Specification Rules." *RFC 5575*, Internet Engineering Task Force.

[11] AT&T, "AT&T and Cisco collaborate to enhance network security and performance," AT&T Newsroom, 2020. [Online]. Available: https://about.att.com/story/2020/cisco_att.html. [Accessed: Jun. 6, 2024].

[12] R. van Rijswijk-Deij, A. Sperotto, and A. Pras, "DNSSEC and its potential for DDoS attacks: a comprehensive measurement study," in *Proceedings of the 2014 Conference on Internet Measurement Conference (IMC '14)*, Vancouver, Canada, November 2014, pp. 449–460. [Online]. Available: https://doi.org/10.1145/2663716.2663731

[13] A. S. M. Rizvi, J. Mirkovic, J. Heidemann, W. Hardaker, and R. Story, "Defending root DNS servers against DDoS using layered defenses," in *Proceedings of the 2023 15th International Conference on Communication Systems & Networks (COMSNETS)*, Bengaluru, India, January 2023, pp. 513–521. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10041415/

[14] M. Allman, "On eliminating root nameservers from the DNS," in *Proceedings of the 18th ACM Workshop on Hot Topics in Networks (HotNets '19)*, New York, NY, USA, Nov. 2019, pp. 1–8. [Online]. Available: https://doi.org/10.1145/3365609.3365863

[15] Root Server Operators, "Root Server Operators," available at `https://root-servers.org/`, accessed on December 6, 2024.

[16] C. Partridge et al., *Host Anycasting Service*, RFC 1546, Internet Engineering Task Force (IETF), Nov. 1993. Available: `https://www.rfc-editor.org/rfc/rfc1546`

[17] Microsoft, "Anycast support in Azure Route Server," Microsoft Learn, [Online]. Available: https://learn.microsoft.com/en-us/azure/route-server/anycast. [Accessed: Jun. 6, 2024].

[18] W. B. de Vries, R. de O. Schmidt, W. Hardaker, J. Heidemann, P.-T. de Boer, and A. Pras, "Verfploeter: Broad and load-aware anycast mapping," *Journal Article*. [Online]. Available: TBD

[19] J. Ginesin and J. Mirkovic, "Understanding DNS Query Composition at B-Root," in *2022 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*, Las Vegas, NV, USA, 2022, pp. 265–270. Available: `https://ieeexplore.ieee.org/abstract/document/10062307/`.

[20] A. Rizvi, "Anycast Agility: Network Playbooks to Fight DDoS" in *31st USENIX Security Symposium (USENIX Security 22)* , August 2022.

[21] A. Rizvi, "Anycast Agility: Network Playbooks to Fight DDoS" in *31st USENIX Security Symposium (USENIX Security 22)* , August 2022.

[22] AT&T, "AT&T and Cisco collaborate to enhance network security and performance," AT&T Newsroom, 2020. [Online]. Available: https://about.att.com/story/2020/cisco_att.html. [Accessed: Jun. 6, 2024].

[23] Root Server Operators, "Events of 2016-06-25," June 2016. [Online]. Available: https://root-servers.org/media/news/events-of-20160625.txt

[24] B-root DNS, "Enabling DNS over TLS on the B-root Server," February 28, 2023. [Online]. Available: https://b.root-servers.org/news/2023/02/28/tls.html

[25] Root Server Operators, "Events of 2015-11-30," Nov. 2015. [Online]. Available: https://root-servers.org/media/news/events-of-20151130.txt

[26] D. Kavitha and R. Ramalakshmi, "Machine learning-based DDoS attack detection and mitigation in SDNs for IoT environments," *Journal of the Franklin Institute*, vol. 361, no. 17, pp. 107197, 2024.

[27] Jones, B., Feamster, N., Paxson, V., Weaver, N., Allman, M.: Detecting DNS Root Manipulation. In: International Conference on Passive and Active NetworkMeasurement. pp. 276–288. Springer (2016)