# Revealing the Architectural Design Patterns in the Volumetric DDoS Defense Design Space

Zhiyi Zhang [1], Guorui Xiao [2], Sichen Song [2], R. Can Aygun [2], Angelos Stavrou [2], Lixia Zhang [2], and Eric Osterweil [2]

[1]UCLA
[2]Affiliation not available

May 31, 2023

# Revealing Protocol Architecture's Design Patterns in the Volumetric DDoS Defense Design Space

Zhiyi Zhang, *Member, IEEE,* Guorui Xiao, Sichen Song, R. Can Aygun, *Student Member, IEEE,* Angelos Stavrou, *Senior Member, IEEE,* Lixia Zhang, *Fellow, IEEE,* Eric Osterweil, *Member, IEEE,*

***Abstract*—Distributed Denial of Service (DDoS) attacks have plagued the Internet for decades. Despite the ever-increasing investments into mitigation solution development, DDoS attacks continue to grow with ever-increasing frequency and magnitude. To identify the root cause of the above-observed trend, in this paper, we conduct a systematic and architectural evaluation of volumetric DDoS detection and mitigation efforts over 24,000 papers, articles, and RFCs over 30+ years. To that end, we introduce a novel approach for systematizing comparisons of DDoS research, resulting in a comprehensive examination of the DDoS literature.**

**Our analysis illustrates a small set of common design patterns across seemingly disparate solutions, and reveals insights into deployment traction and success of DDoS solutions. Furthermore, we discuss economic incentives and the lack of harmony between synergistic but independent approaches for detection and mitigation. As expected, defenses with a clear cost/benefit rationale are more prevalent than those that require extensive infrastructure changes. Finally, we discuss the lessons learned which we hope can shed light on future directions that can potentially turn the tide of the war against DDoS.**

***Index Terms*—DDoS, Internet Architecture, DDoS Defenses, DDoS Design Patterns.**

## I. Introduction

DISTRIBUTED Denial of Service (DDoS) attacks have plagued the Internet for over 20+ years [1]. In the decades that the DDoS problem has been well established and studied extensively, billions of dollars have been invested in defenses, and even more billions of dollars have been lost due to DDoS-induced outages. Figure 1 illustrates that more than 20-years worth of research literature on DDoS mitigation, composed of thousands of articles, papers, Requests For Comments (RFCs), and patents of inventions, is sizable and continues to grow linearly with time. Meanwhile the DDoS attack traffic volumes and amount of money spent in defending against them are growing at a super-linear rate. These strictly empirical measures suggest that the DDoS problem is worsening, despite our continued efforts to address it.

Zhiyi Zhang, Guorui Xiao, Sichen Song, R. Can Aygun, and Lixia Zhang are with the Department of Computer Science at UCLA (email: zhiyi@cs.ucla.edu; grxiao@ucla.edu; songsichen@ucla.edu; rcaygun@cs.ucla.edu; lixia@cs.ucla.edu)

Angelos Stavrou is with the Bradley Department of Electrical and Computer Engineering at Virginia Tech (email: angelos@vt.edu)

Eric Osterweil is with the Department of Computer Science at George Mason University (email: eoster@gmu.edu)

This work is partially supported by the National Science Foundation under award 1719403

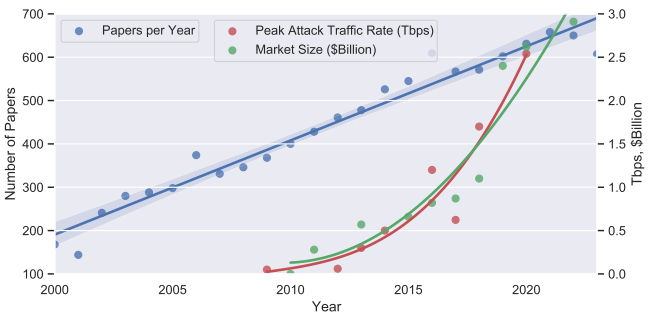


Fig. 1. DDoS paper publication (from the corpus described in Section IV-A) from 2000 to Oct. 2023, market size from 2010 to 2022 [2]–[5], and scale of famous attacks [6]–[8]

Our goal in this paper is to understand the fundamental reason(s) why decades of investment have not resulted in winning the DDoS war. Where are we now with respect to winning the war? And where might we be 20 years down the road? Although these questions may sound too big to answer, avoiding them is not a winning strategy. "The farther back you can look, the farther forward you are likely to see" – assuming Churchill is correct, this paper takes the first step towards understanding where we are now in mitigating DDoS, through a systematic examination of past efforts.

To make this first step attainable, we note that DDoS attacks have evolved into different types: some aim to exhaust resources, ranging from bandwidth in network links and devices to computation and storage capacity in applications; while others exploit vulnerabilities in Internet infrastructure, network protocols, or application systems. Therefore, in this paper we focus on surveying mitigation efforts against a specific class of DDoS attacks, which deny service through overwhelming volumes of network traffic, known as "volumetric attacks." Volumetric DDoS attacks have a relatively simple success criterion: to inundate destinations (*i.e.*, "victims") with large volumes of traffic overwhelming their systems and/or exceeding their network(s) capacity. The largest DDoS attacks seen to date have been launched from *distributed* robot (bot) networks (or botnets). Further, volumetric DDoS attacks are also the subject of the majority of DDoS publicity and mitigation literature. We derive our findings from a comprehensive critical examination of the sizable corpus of literature that addresses volumetric DDoS by conducting a systematic analysis of the landscape of both proposed and deployed solutions.

Specifically, we systematize the selection of the most im-

pactful representatives from all the published works by using their number of citations as a metric. Since more recent publications may not have received high citations, we include papers from recent conferences with high impact measures. We then classify all the DDoS defense solutions by their shared functional approaches, articulate their effectiveness, and identify barriers in their deployments.

Developing this paper has been a long learning process for ourselves: we examined the large number of different solutions, sorted and resorted them between different categories, went back and forth between understanding what each solution proposes and how to articulate its deployability, and then systematically characterized the examined solutions.

**Contributions** We observed that most proposed DDoS remediations appear attractive but stay at "being proposed" stage, because they require coordinated network infrastructure upgrades but do not provide sufficient incentives to result in adoption. In contrast, DDoS remediations that are widely deployed today are managed by *single parties*, employ the existing infrastructure as is, and form overlay solutions for DDoS mitigation. These basic observations essentially use the market as an evaluation technique and form the rigorous foundation on which our classification schema stands. Moreover, while each piece of research literature adds to the collection of new ideas and enhancements, they rely on a comparatively small set of *design patterns* to mitigate volumetric DDoS attacks. These design patterns are used to remediate a similarly small set of *fundamental properties* of the IP network architecture that make volumetric DDoS attacks easy to launch but difficult to defend (Section II). The descriptions of the design patterns that we can distill from the literature and our assessment of the solution-gaps between proposals are summarized in Section VI, together with our postulation on future DDoS mitigation directions. In summary:

- We develop a classification schema (Section IV) which serves as an organizational guide for systematizing comparisons for DDoS research. Using this schema, we conduct a comprehensive examination of the DDoS literature spanning more than 24,000 papers, articles, and RFCs, derived from Google Scholar [9], [10] and DBLP [11] from 30+ years.
- Our analysis of the literature provides insight into which aspects of DDoS solution proposals and systems correlate with successful deployment traction, and illustrates common design patterns across seemingly disparate solutions. One fundamental design pattern that emerges repeatedly across the solutions is to enshrine *state* into stateless IP forwarding. In order to differentiate regular from attack traffic, the proposed solutions either add the state to the deployed IP routers, or carry the state in IP packets, or otherwise direct traffic to *overlay networks* to sort out. The first two require changes to the deployed Internet infrastructure, while the third one bypasses such a deployment hurdle.
- Our analysis illustrates that some design patterns rise in popularity across multiple papers in the literature in temporal clusters, and are then succeeded by other approaches. For example, Figure 2, illustrates the evolving popularity of various design patterns over the years, by measuring the frequency of associated keywords within the titles of the surveyed papers. This creates the impression that certain approaches (along with their referenced works) are outdated, but this observation is linked to the dynamic shifts in research themes within the literature rather than their real world deployment. Indeed, the evolution of research topics is largely detached from their practical deployment, as none of the popular topics have been successfully implemented at scale in the real world.
- We provide insights into the deployment challenges and incentives of proposed solutions and the lack of harmony between synergistic but independent approaches to detection and mitigation. Single party solutions with clear cost/benefit incentives have been deployed today while solutions that require multi-party coordination for large infrastructure changes have not. Based on our observations, we further discuss future DDoS mitigation directions in the long run.

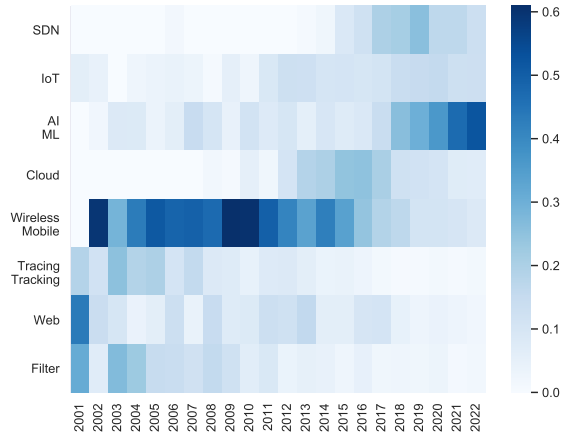


Fig. 2. Evolving popularity of certain design patterns after 2000. Topic keywords are extracted from surveyed papers, utilizing terms from index terms and titles. The heat map values are determined by the ratio of papers of a specific topic to the total number of papers across all selected topics.

**Outline** The paper is organized as follows: Section II offers a succinct summary of the basic properties of the TCP/IP architecture that make DDoS attacks easy to launch but difficult to defend. Section III compares our work with the previous survey papers on DDoS. Section IV describes our methodology and how we classify the large corpus of literature on DDoS mitigation. Section V examines the different classes of work in detail. Section VI uses an architectural lens to summarize the existing DDoS solutions and apply the lessons learned to other types of DDoS attacks. Finally, Section VII concludes the paper.

## II. DDoS Exploits Properties of TCP/IP Networks

To understand why DDoS attacks have been (and continue to be) a perennial sickness in cybersecurity, we begin with a principled inspection of specific properties in IP that make DDoS attacks easy to launch, and make effective remediations difficult. Next, in Section II-B, we illustrate how those basic

properties enable DDoS attacks. We then articulate why effective remediation is still elusive by pointing out the functional support that is missing in the IP design, yet necessary for effective remediation in Section II-C.

### A. IP's Basic Properties

IP's 40+ year old design has the following basic properties:

1) Any host $h_1$ can send packets to any other host $h_2$, as long as $h_1$ has $h_2$'s IP address (push unsolicited traffic).
2) IPv4 address space is relatively small in size. Modern tools like ZMap [12] can *entirely* scan through it in a short time.
3) The TCP/IP protocol stack was designed without security.
4) IP routers forward packets based on their destination addresses *only*, *i.e.*, there is no source address validation.
5) Routers' forwarding plane has no state, and treats all packets equally.
6) Its operated as an interconnection of autonomous networks, whose only cross-administrative coordination is peering to propagate routing updates.

### B. How IP's Design Makes DDoS Easy

While the objective of botnet volumetric DDoS attacks is straightforward, they often involve several stages and a control infrastructure, as shown in Figure 3. The first step in launching a botnet-based DDoS attack is to establish a botnet. Bots are devices or hosts that have been compromised by a miscreant, who then has full/partial control over their actions. This first step is made easy by the *IP properties 1-3* (defined above in Section II-A). These inherent properties of IP allow a miscreant to easily enumerate the entire IP address space to find (vast swaths of) vulnerable hosts by sending unsolicited probes, as described in the literature [13]. Then, vulnerable hosts are infected and registered into attackers' report infrastructures. Collections of compromised bots are "herded" together into botnets by "herders", who coordinate their botnets through their own Command and Control (C2) infrastructures. Attack commands can then be issued through C2 infrastructure, and the damage of a volumetric DDoS attack comes from the *aggregate* volume of bots' capacities.

In addition, *IP property 4*, from Section II-A, is responsible for some of the largest DDoS attacks ever recorded using an additional technique called reflective amplification. In those attacks, numerous bots send seemingly legitimate application queries to large infrastructure services, with false (or "spoofed") source addresses set to a victim's address. The attackers, thereby, "reflect" queries off one or multiple service providers toward a single victim, as described in the literature [14]. Worse yet, these attacks capitalize on services where responses can be much larger than queries (*e.g.*, DNS, SNMP, NTP [15], `memcached`, TCP queries to nation-state censor middleboxes [16], etc.). These larger responses further "amplify" the aggregated query traffic volume, and can be from $\times 2$ to up to $\times 1,000$. These volumes are the aggregate of the attacking botnet's total capacity $\times$ the amplification factor.

The above illustrates which underlying *properties* of the IP architecture enable DDoS attacks: allowing anyone to push traffic to anyone else, finite address space, unprotected end devices (hosts), and no source address check. Next we discuss the IP's properties that hamper remediation solutions.

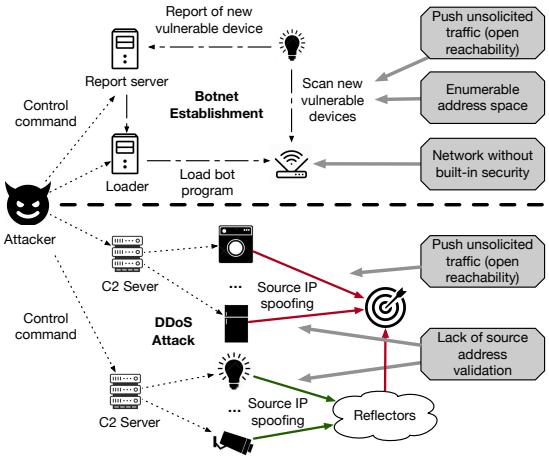


Fig. 3. IP design makes DDoS botnet establishment and attack easy

### C. How IP's Design Makes DDoS Defense Difficult

*IP property 4*, from Section II-A, confounds DDoS remediation by allowing bots to spoof *source* IP addresses in their attack traffic (making the traffic seem to come from other hosts), as described in the literature [14]. Fundamentally, in the IP protocol architecture, one cannot squelch an attack source that cannot be identified. Further, the IP protocol has stateless packet forwarding but an essential goal of DDoS detection and remediation approaches is to detect and classify good from bad traffic. Repeated design patterns in the literature suggest this requires packet and/or flow state. In some approaches (described in Section V), the network layer infrastructure (routers), or overlay servers, must be enhanced to maintain additional state, and in others state is added into the packets and/or flows to address the missing features listed in IP's basic *properties 5 & 6*.

Our literature survey indicates that there is an emergent set of common *design patterns* that appear across most proposed DDoS defenses and deployed solutions. Moreover, some of these patterns reveal additional remediation properties that are needed (but are missing) from the original IP architecture. Another fundamental consideration that underlies the bulk of approaches discussed in Section V is *where* changes must be by made, and by *whom*, for the deployment to succeed.

## III. Previous Assessment Attempts

Over the past twenty years, many review and survey papers on DDoS attack and mitigation mechanisms have been published [17]–[52]. Assessments and classifications have ranged from inspecting specific tools used in attacks to high-level approaches used in defense mechanisms.

Before 2000, an early work authored by Howard *et al.* [17], [18] proposed a taxonomy of network attacks in general, where DDoS attacks were not specifically highlighted because they were not a big threat at the time. From 2000 to 2010, Kargl *et*

*al.* [19] presented a classification of DDoS attacks based on the victim type (*e.g.*, web servers, routers, middleboxes), the resource that is being exhausted, and the vulnerabilities exploited by the attack. A tutorial [20] introduced the concept of an *Internet firewall* and organized DDoS attacks into prevention/preemption, source identification/traceback, and detection/filtering. Hussain *et al.* [23] classified DDoS attacks based on the botnet size generating traffic and whether the traffic is reflected. Mirkovic *et al.* [24] presented taxonomies for both DDoS attacks and defenses, in which the defense mechanisms are classified by the activity type, coordination level, and deployment location. A survey [37] classified DDoS defense mechanisms into: common countermeasures, statistical approaches, and traceback. Each category was sorted into three branches. Antonakakis *et al.* [53] studied the Mirai botnet. Harris *et al.* [38] and Griffioen *et al.* [52] performed "kill chain" analysis on DDoS attacks. Feily *et al.* [30] surveyed botnets and classified botnet detection mechanisms.

Other survey and review works focus on specific types of DDoS attacks or defense mechanisms in certain types of environments. Specifically, reviews [40]–[43], [45], [47], [54] concentrate on DDoS attacks and mitigation in the cloud computing environment. Reviews [40], [41], [46], [49] focused on the DDoS attacks on, and defenses for, Software-Defined Networks (SDNs). Reviews from Douligeris *et al.* [25] and Rao *et al.* [50] focus on DDoS detection utilizing artificial intelligence and statistical approaches. The review from Praseed *et al.* [55] is specific to the application layer DDoS. Also, previous works [21], [22], [28], [33], [34] focus on the DDoS attacks and defenses in wireless networks and Wireless Sensor Network (WSN).

Notably, Peng *et al.* [27] explain how the Internet's architectural design can affect the DDoS attack and defense. The authors summarize these principles as (i) resource sharing (*e.g.*, Internet's packet switching v.s. telephone network's circuit switching), (ii) simple core and complex edge (*e.g.*, sophisticated solutions, such as packet authentication, cannot be deployed in the Internet core), (iii) asymmetric routing (*e.g.*, make IP traceback harder), (iv) fast core and slow edge network (*e.g.*, edge networks can be easily overwhelmed), and (v) decentralized Internet management (*e.g.*, make large scale deployment harder). This paper, published in 2007, is the closest to our work, but we contend that a more principled analysis, together with a more up to date and comprehensive review of the existing work, is needed to lay a path forward.

Our goal in this paper is to (i) understand how the underlying IP architecture impacts both DDoS attacks and defenses, (ii) identify common features extracted from all the existing DDoS defense approaches, and (iii) associate those features with their required architectural changes, to understand why most of the proposed solutions have not been deployed, as well as the essential properties of the small set of deployed mitigation solutions.

## IV. Classification Methodology & Rationale

### A. Methodology and Corpus Collection

To understand the evolution, evolutionary forces, and even the precursor events that have led to DDoS attacks, it is paramount for the corpus of surveyed material to be both complete and representative. To that end, we collected about 24,000 papers from Google Scholar [9], [10] and DBLP [11] from 1980 to 2021 that mentioned DDoS-related keywords in their title and content. Manuscripts from as far back as 1980 predate the canonicalization of DDoS but serve as precursors to DDoS. The keywords are extracted by DDoS-related terms, including *DoS, DDoS, denial, denial-of-service, flooding, capability, filter, filtering, botnet, blackholing, and scrubbing*. After filtering out the papers whose main focus is not on denial of service attacks or defenses, we narrowed the list to approximately 8,500 papers for further examination. Subsequently, we ranked them based on their citation-counts and average citation counts per year (high to low). Next, we carefully reviewed the first 250 papers manually. In this process, we first filtered out 47 papers that are out of the scope of this paper, *e.g.*, application-layer DDoS attacks, physical layer signal jamming, and botnet identification/mitigation. Then we added 40 papers that were frequently cited by the remaining papers to our corpus; for example, some papers do not frequently mention the DDoS related keywords but can be used for DDoS mitigation, such as "off by default" [56] and "Controlling high bandwidth aggregates in the network" [57]. In addition, our corpus also includes more than 30 review/survey papers, which are discussed in Section III. Furthermore, since more recent publications may not have received representative citation index, we manually searched for the relevant studies from the highly cited security and network conferences (*e.g.*, IEEE S&P, Usenix Security, ACM CCS, and NDSS) from 2020 to 2022 and added 11 more papers into our corpus. Finally, we investigated the DDoS signaling RFCs [58] developed by IETF and ended up with 264 papers in total. The full list is available online at [59].

In order to emphasize the systematization rather than inserting a full catalog in-line, we picked representative approaches for each category so that less than 2/3 of the 264 papers are directly referenced in this paper.

### B. Classification Rationale

Different from conducting a survey, our primary goal is to perform a methodical examination of the commonalities and differences, with a primary focus on characterizing different approaches developed by both researchers and practitioners. We classify DDoS defenses solutions into four candidate categories based on our intuitive view on the different stages in the process of mitigating network DDoS threat: preventing DDoS, detecting DDoS, mitigating DDoS, and holistic solutions. We provide retrospective analysis on this initial intuition in Section VI. By deepening our understanding of where these methods succeeded in pushing the state of the art, and where they fell short (in particular in their deployability), we hope to identify the true challenges in DDoS mitigation and, consequently, where effective solutions lie.

Our **Preventive** approaches category attempts to address network-level preventive remedies, ranging from conformity with standards to traffic by permission only. The solutions in this category perform no active measurement for DDoS

detection. We place the approaches that attempt to detect DDoS by performing measurements in the **Detection** category; some solutions also mark offending flows or packets. However, all the works in this detection class assume that a separate mitigation mechanism is in place to alleviate the DDoS attack upon detection. In the **Mitigation** category, the focus of the proposed solutions is "quashing" the DDoS attack once detection mechanisms have raised the alarm. All the mitigation approaches assume the presence of a detection mechanism, together with some packet or flow-level information that can be used to filter out or redirect traffic. Defenses that address DDoS attacks by providing combined detection and mitigation solutions are placed in the **Holistic** defenses category.

Our goal in this paper is to reveal the underlying characteristics of the proposed solutions and how they attempt to remedy the inherent shortcomings inherent in the IP design. We derived this candidate classification rubric by using deployment and market presence as an evaluation tool that illustrates one measure of what holistically "works." We present this approach as one candidate way in which classification can be done, and other valid approaches are certainly possible. We examine each proposed solution and identify its answers to the following questions:

1) *Where* is the proposed solution being deployed (*e.g.*, end hosts, edges, routers)?
2) *What* type of changes are required to deploy the proposed solution?
3) *Who* needs to make the change? In particular, can the proposed solution be effective if deployed unilaterally by a single party? Does the proposed solution provide (economic) incentives to first movers?
4) If coordination among multiple parties is required, they need to establish security/trust relations. *How* is that set up, or *how* is it assumed to pre-exist, and *when* in the DDoS mitigation process does it operate? To represent this across the diverse proposals and techniques, we summarize what *cardinality* is necessary for trust. That is, if one party needs to trust $n$ other parties, we denote this as "$1 : n$". Alternately, if $n$ parties need to trust $m$ other parties, we denote this as "$n : m$".
5) Does this solution share any commonalities with other approaches? What are the novel design elements?
6) Are deployment costs aligned with benefits, *i.e.* does a deployer reap benefits from making the required investment?

For completeness, our full systematization is available at [59].

## V. Understanding Existing DDoS Defense Mechanisms

The terminology used in the literature is not consistent. For coherence in our analysis, we use the following terminology in this paper. **Senders** denote traffic sources/clients, benign or malicious. **Receivers**, as the target of DDoS attacks, are the destinations of traffic from the senders. **Endhosts** refer to both senders and receivers. **Upstream** is towards the sender and **downstream** is towards the receiver.

### A. Preventive Approaches

This class of proposed DDoS mitigations does not *actively* detect DDoS using measurements or situational awareness. Instead, they apply a set of predefined network compliance rules to fence off unwanted traffic (see Section IV). Based on their designs, we divide the preventive solutions into the following three broad categories: (i) Discarding packets carrying spoofed source addresses and non-conforming traffic, categorized below in Section A.1; (ii) Letting receivers issue permissions to senders either at the network layer (*e.g.*, in the form of tokens), or the application layer (*e.g.*, proof of work), detailed in Section A.2; and (iii) Adding new semantics to IP addresses, or new routing scope control to limit receivers' reachability, summarized in Section A.3.

**A.1 Preventing Address Spoofing & Non-conforming Traffic** This type of preventive approach drops packets with spoofed source address [60]–[69]. They utilize network topology and host connectivity to identify non-conforming packets.

**A.1.1 Simple Filtering at Edges** Ingress filtering (BCP 38) [60] lets stub networks, or their immediate provider, deploy a packet filter at its exit points to drop all outgoing packets whose source IP addresses do not belong to the local Internet Service Providers (ISPs). This action does not need coordination with other parties and is similar to egress filtering on source address [61], [64]. A universal deployment of BCP 38 would eliminate traffic with spoofed source addresses, thus preventing simple reflection attacks. For example, it has been listed as one of the main action items in a global initiative called Mutually Agreed Norms for Routing Security (MANRS) [70]. However, an adopter of BCP 38 helps drop attack traffic to others with no direct benefits to itself, and a recent study shows that 69.8% of ASes on the internet do not perform ingress filtering [71]. Such limited deployment of ingress filters has little effectiveness in curtailing reflection attacks, which not only causes untraceable DDoS attack senders, but also makes sophisticated remote off-path DoS attacks such as ICMP redirection based blackholing [72] and DNS cache poisoning [73] possible. The lack of incentives for stub networks to adopt BCP 38 led to solutions that perform filtering at the network core.

Systematizing from Section IV-B, the "where" of this class is at *routers*; the "what" is *none* (local configurations needed, only); the "who" only needs to be *sender networks*; for the "how," trust is *not needed*, as configurations are local to routed resource holders; the novelty of this approach is that it prevents attack traffic from being admitted to the Internet; and "cost/benefits alignment" is *fully misaligned*, as beneficiaries cannot deploy protections and deployers gain no direct benefits.

**A.1.2 Router State based Filtering** Multiple solutions were proposed to provide routers with required information to filter out non-compliant traffic [62], [63], [66]. SAVE [63] builds a new incoming traffic table at each router to specify the valid source address space for each interface. Each SAVE router is associated with a set of *source addresses* and periodically generates SAVE updates and sends them to each entry in its forwarding table. SAVE updates associate valid source address

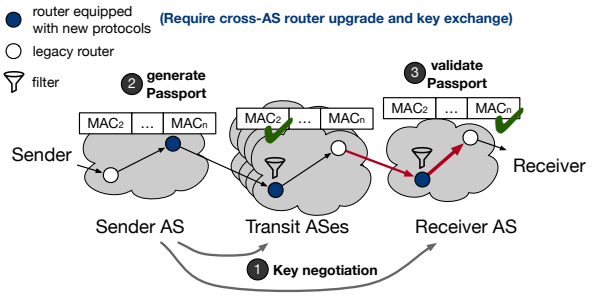


Fig. 4. Passport [66] (cited by 220+) puts state in packets and requires inter-AS key exchange.

blocks with incoming interfaces at routers along the paths, *i.e.*, they set up *new state* required for packet filtering. However, because IP routers maintain a Forwarding Information Base with `[prefix, outgoing interface]` mapping only, deployment of SAVE requires modifications to a large portion of the routers in the network.

Systematizing our analyses, the "where" of this class is at *routers*; the "what" is *router upgrade for state and enforcement*; the "who" is *on-path transit ISPs* (which can be cadres of ISPs instead of global deployment); for the "how," state maintained in routers is used for enforcement; the novelty of this approach is it uses in-network processing to shed DDoS traffic before it aggregates; and "cost/benefits alignment" is *fully misaligned*, as beneficiaries cannot deploy protections and deployers gain no direct benefits from deployment.

**A.1.3 Packet State based Filtering** To address the challenges of setting up and maintaining *new state* at routers, Passport [66] lets individual packets carry the information needed for their source address validation. It attaches the AS path to each packet with cryptographic protection (Figure 4). The border router of the sender AS, denoted by $R_S$, adds a series of Message Authentication Codes (MACs) to the outgoing packet for each AS, denoted by $A_i$, along the path to the destination. Each MAC is generated with a shared secret between a ($R_S$, $A_i$) pair (❷), enabling ASes along the path to verify MACs using corresponding shared secrets with the AS (associated with the packet source address) to eliminate spoofing (❸). To generate these MACs, all the other ASes must share a secret key with the sender AS beforehand (❶). Passport requires both cross-AS trust establishment and upgrades of ASes' border routers to perform the MAC verification for each packet.

Systematizing our analyses, the "where" is at *routers*; the "what" is *adding state to packets and router upgrade* for enforcement; the "who" is *on-path transit ISPs* with established secure coordination; for the "how," state maintained in packets and any upgraded routers can be used for enforcement; the novelty of this approach is it uses partial in-network processing to shed DDoS traffic before it aggregates; and "cost/benefits alignment" is *fully misaligned*, as these solutions do not bring first-mover benefit to individual ASes.

**A.2 Using Packet Tokens for Access Control** Here, we discuss access permissions using capability tokens and permissive access control by packet marking that can be used later to discard attacking traffic. We do not cover the application layer access control mechanisms such as puzzle-based approaches [74]–[78] since they are less related to the Internet architectural designs.

**A.2.1 Communication Access using Capabilities** This type of solution requires each sender to obtain its permission (capability) from the receiver first, enabling explicit receiver authorization for traffic [79]–[83].

Anderson *et al.* [79] was first to propose that a sender must send a request-to-send (RTS) to a receiver for permission before sending traffic. To prevent RTS packets from becoming attack vectors, [79] proposed an overlay of RTS servers in all ASes. The overlay network would carry/regulate RTS packets to prevent RTS DDoS attacks. To accept traffic from a specific sender, a receiver would respond with a chain of hash values as tokens with limited packet budget and time duration used by the sender. ASes along the path serve as Verification Points (VPs) to filter unwanted traffic by maintaining the *token state for each flow*, thus making stateless IP routers stateful.

Yaar *et al.* proposed Stateless Internet Flow Filter (SIFF) [80] which lets packets carry the state information using verifiable tokens instead of maintaining router state. Senders and receivers obtain capability tokens via a handshake, which is carried in packets. Routers along the path can validate independently. SIFF can be combined with puzzle auctions to prevent token-generation resource exhaustion. SIFF still requires cross-AS router upgrades to examine the packets and verify if the capability is granted. To initiate communications, the sender sends an EXPLORE packet allowing routers along the path to insert markings into a header field.

In TVA [81], [83], Yang *et al.* observed several weaknesses in SIFF, including (i) the capability field is short and thus potentially subject to brute-force attack; (ii) SIFF has no per-flow state of the capability status, attackers can circumvent the handshake step by replaying valid packets; and (iii) attackers can establish approved connections among themselves and flood the network to drown out EXPLORE packets from legitimate users.

To address these issues, TVA enlarges the capability field and binds each capability to a specific path and time period using a cryptographic MAC derived from the sender and receiver's IP addresses, the network path, and a timestamp. Routers need to maintain per-flow state to verify the capability against the time duration and allowed amount of traffic. Second, to prevent router memory exhaustion, TVA keeps state only for flows whose sending rate is above a pre-defined threshold. Third, TVA leverages fair queuing to reduce the impact of denial of capability request (*e.g.*, RTS flooding, EXPLORE flooding). TVA requires routers at trusted boundaries (*e.g.*, ASes) to coordinate in maintaining a slow-changing secret key for MAC generation and maintain flow state for high rate flows.

To reduce the impact of DDoS against capability requests, two methods were proposed: building an overlay to issue capabilities and utilizing fair queuing. These approaches moved the DDoS challenge from defending the victim receiver to defending the infrastructure that issues the capabilities.

Systematizing, the "where" is either at *routers*, in *packets*,

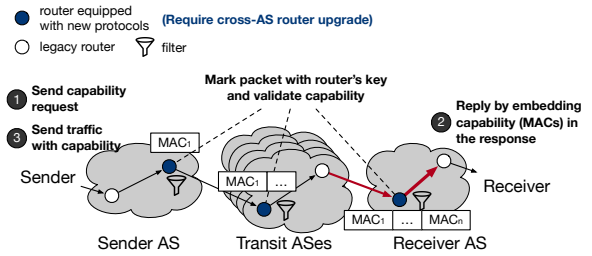


Fig. 5. TVA [81], [83] (Cited by 660+) utilizes packet marking as the granted capability to senders

or in *both*; the "what" is *router upgrade* (processing in routers and state in packets); the "who" is *on-path transit ISPs* (either global or cadres of on-path ISPs); the "how" is *packet state* is annotated and *routers enforce*; the novelty is that state is kept in packets and authorization is precomputed at endpoints; trust is $1:n$, from all senders to a receiver; and "cost/benefits alignment" is *misaligned*, as on-path router operators do not benefit from deploying and enforcing.

**A.2.2 Enabling Path Traceback using Tokens** The prevalence of IP Spoofing and IP's stateless forwarding plane complicate the attack sender identification. Multiple papers [84]–[89] attempt to address this problem by keeping the path state in routers or packets. Different from capability-based solutions, this class of defenses simply provides useful information for traffic classification, to be used later in DDoS mitigation when/if needed. Some examples using similar tokens for filtering are discussed in Section V-C.3.1.

Pi (Path Identification) [89] and packet marking based IP traceback works [85], [86] let network routers mark packets passing by, making each packet carry some *path state*. In Pi's case, the state is a list of hashes of the routers' IP addresses, serving as an identification of the traffic path. Pi's marking mechanism is similar to that shown in Figure 5 but is simpler because using IP addresses as markings requires no key exchange or cryptographic operations. Another example is the router-based IP traceback mechanisms proposed by Snoeren *et al.* [87], [88]. Instead of marking packets, they let *routers keep state* for each passing packet so that the path information for a given packet can be extracted out later in a per-hop manner.

Systematizing, the "where" of this class is *routers and packets*; the "what" is *router upgrade* for processing and *state in packets*; the "who" is a *on-path routers* (transit); the "how" is *routers annotate packets*; this approach's novelty is simplicity and reduced coordination/trust; "trust configuration" is *not needed*; and "cost/benefits alignment" is *misaligned*, as beneficiaries cannot deploy protections and deployers gain no direct benefits from deployment.

**A.3 Access Control via Controlling Network Reachability** This class of solutions change the IP address semantics or existing routing system to allow receivers to control their reachability [56], [90]. Handley *et al.* [90] propose to explicitly separate the IP address space between clients (senders) and servers (receivers). Following the design, only clients can initiate connection to servers, reducing the spread of botnet and reflection attacks. To prevent address spoofing and reflection attack, symmetric paths between clients and servers are enforced via path-based addressing: client $C$'s packet going to server $S$ will have its source address $C_A$ prepended with each passing domain's ID. When a response comes back, each domain on the path can verify its own ID, then remove it from $C_A$. However, this design is limited to stationary clients, as the $C_A$ for a mobile client may change as the mobile moves during a connection and the same holds for in-network path changes.

Another work [56] proposed by Ballani *et al.* lets receivers set their reachability to be "off" by default. When a receiver $R$ wants to become reachable, it sends reachability advertisements to the routing system with a set of constraints attached. These receiver-specified constraints may state $R$ is "on" to all/selected hosts, or "off" to specified hosts. Routers aggregate reachability announcements from end hosts and then propagate them to the Internet. Such reachability restrictions can either be proactive (off by default), or reactive upon attacks (in this case, the solution becomes a Mitigation approach in Section V-C). Routers along the data path must check the reachability table, dropping packets when the reachability of the packet's destination is off. Deployment requires a coordinated upgrade of routers globally to support the new reachability protocol and maintain new state in lookup tables.

Systematizing, the "where" of this class is at *routers* and *endhosts*; the "what" is *address renumbering* and *router upgrade*; the "who" is *global*, all routers in the Internet; for the "how," destinations disseminate reachability with authorization; the novelty of this approach is it eliminates attack surface; the "trust coordination" is *m:n*, global routed resource certification needed; and "cost/benefits alignment" is *misaligned* because it requires an "Internet flag-day" that may not benefit all users.

**Summary** The basic changes (*i.e.* "how") of solutions in A.1 is to add new state to routers to eliminate traffic with spoofed source addresses; the solutions "how" in A.2 changes IP's "any node can send packets to any other node" model to transmit-by-permission from *receivers* that also require router support; and the "how" of A.3 needs fundamental changes to today's IP addressing and forwarding.

The "novel" shared feature among the solutions in A.1 and A.2 are setting up control state on the *data plane* to sort out "allowed" from "disallowed" traffic, a big departure from IP's stateless data plane. Solutions principally differ in the type and amount of state to keep, and whether the state is stored in routers or carried in packets. One solution places all the needed state information at routers [79], the others [91], [66] make packets carry the state information to reduce the burden at routers. Finally, for our "cost/benefits alignment" systematization, all of the solutions from A.1 to A.3 misalign costs and benefits, which undoubtedly plays a role in their deployability, because first movers get no direct benefits. This challenge frames a dilemma: there may be no first movers if they do not benefit directly and immediately, and there will be no wide adoption without first movers.

### B. DDoS Event Detection and Traffic Classification

In this section, we analyze approaches that attempt to detect volumetric DDoS attacks and "raise-the-alarm" to activate mitigation solutions. It is noteworthy that 45% of the 250+ reviewed papers propose detection-only solutions, and the majority (>60%) employ data mining and/or machine learning.

Across all reviewed works, the detection is carried out by using situational awareness from certain vantage points, ranging from a single network node to multiple routers and ASes. In addition, a subset of the solutions also perform binary classification between attack and benign traffic, to separate "good" from "bad" packets and/or flows. Their metrics for effective attack detection include high accuracy and low latency. Generally, their designs contain the following steps:

1) *Model/Threshold generation* produces the model or threshold to be used for detection or traffic classification. This step is usually done beforehand or offline.
2) *Traffic sensing* employs individual devices, residing either inside the network or at endpoints, to collect traffic information by examining the header, or even the payload, of passing by packets.
3) *Detection step* uses the collected traffic information against the model/threshold generated by Step-1 to detect DDoS attacks and differentiate benign from malicious traffic; the model and threshold may also be updated in this step.

We first focus on *what information*, and *where/how* it is being collected to determine if any infrastructure modifications over the existing Internet are required. We then categorize the representative works by where they collect the information, *i.e.*, near the receiver (Section B.1), near the edges/senders (Section B.2), and distributed across ASes (Section B.3). We do not mention SDN-based detection approaches here since they can be deployed at any of the three location categories. However, the deployment location trade offs may apply to them. For these proposals we provide the systematization of detection approaches in the summary section.

**B.1 Near-Receiver Traffic Sensing** DDoS detection in a receiver's network offer higher accuracy due to increased attack traffic concentration around the victim [92]. It is more tractable to collect information in small scope near the receiver when compared to large-scale deployment that may require coordination among ASes. Further, the receiver and its network provider likely have incentives to deploy detection and traffic classification systems for their own benefits [92].

On the flip side, handling a large onslaught of attack traffic requires intensive computation and memory. Even when the detection is feasible, the proximity to the receiver leaves little time for reaction and has high collateral damage (*e.g.* oversubscribed network paths). Moreover, DDoS mitigation requires real-time information exchange between the receiver, its first hop ISP(s), and ASes along the path, and assumes that the detection and classification results can be trusted to be acted upon.

Yu *et al.* [93] employs a support vector machine [94] classifier on coarse-grain traffic measurements using SNMP for attack detection. The measurements include total packet counts, TCP resets, and other traffic events within a given time window. The coarse granularity of SNMP data limits the classification performance even though the protocol is widely supported. Other approaches aim to achieve higher accuracy by using *per-flow* traffic information (*e.g.*, source/destination IP addresses, protocol with entropy [92]), or even *packet-level* information (*e.g.*, per-packet information from a moving window with deep-learning [95]) but their high resource requirements reduce the routers' forwarding capacity.

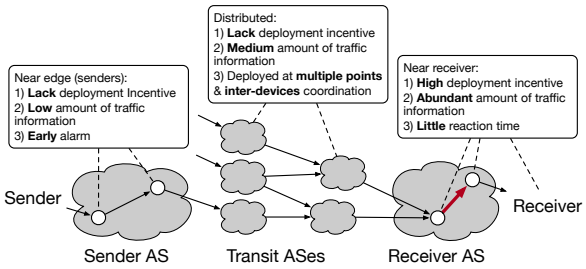


Fig. 6. Information collection for DDoS detection and traffic classification

**B.2 Near-Edge Traffic Sensing** Approaches in this class make sophisticated traffic processing more feasible [96] and they can be coupled with mitigation to limit the attack traffic from exiting the originating networks, reducing collateral harm [97]. However, the detection accuracy is reduced due to the local traffic views used. Moreover, without coordination across ISPs, near-edge detection can be side-stepped by a sufficiently distributed attack that is not as apparent at each sender, and creates an aggregate effect only as it approaches the victim network.

D-WARD [96] collects information by counting packets of each TCP, UDP, and ICMP flow at the edge router of the senders' network. The traffic information is then fed into a model that thresholds the sent/received ratio, concurrent connection counts, and packet rate to detect abnormal flows. To minimize the state overhead, D-WARD uses a *least frequently used hash table* to only keep the state for highly active flows. Since attack traffic can be launched from anywhere, all the near-sender traffic sensing solutions require wide adoptions by edge ISPs in order to be effective.

**B.3 Distributed Traffic Sensing** This set of approaches focuses on utilizing information collected from multiple network locations with cross-domain communication to improve DDoS detection and traffic classification accuracy (though communication and data aggregation may introduce long latency).

Yu *et al.* applied an entropy-based approach to classify attacking traffic [98]. This work identifies each flow by the upstream router and the receiver address. It lets routers maintain the information entropy of network flows and then applies thresholds to detect DDoS events. Since DDoS traffic is most noticeable near the receiver, whenever a DDoS event is detected, downstream routers can inform upstream routers of the event, so that the latter can adjust their thresholds to detect the attack flows and take actions. DCD [99] proposes to utilize a centralized server that collects abnormal traffic information from upstream routers to form a tree. Moreover, to avoid the centralized server, FireCol [100] lets routers

in different ASes directly communicate with each other to detect DDoS events. Distributed traffic information exchanges among routers enables the aggregation of information from multiple viewpoints to reduce false positives. In Section V-D, we discuss signaling approaches for coordination of DDoS detection.

**Systematization and Summary** Figure 6 shows a summary of different information collection strategies used by approaches in this category. We observe two common design patterns: our systematized "who," "what," "where," "how," and the novelty of these approaches are essentially identical. The "where" of these approaches is *at routers*, the "what" is *router upgrades*, the "who" are *on-path transit routers*, "how" is performing *ML on locally observed traffic* on per-packet (*e.g.*, [93]) or per-flow (*e.g.*, [95]) telemetry. The novelty of approaches in B.1 and B.2 is primarily to drop traffic identified by ML. B.3 adds to this by introducing a framework to include distributed observations, identifying that there is a benefit in collecting network data as close to attack originators as possible for early alarms. However, without timely coordination across ASes, detection solutions at the edge have a limited view of traffic, especially for multi-homed servers. The effectiveness of detection based on partial information is reduced for distributed attacks. Moreover, the desired high accuracy severely increases the router overhead due to the need for finer granularity of traffic state. Regarding "trust coordination", whereas B.1 and B.2 do not accommodate distributed coordination, B.3 requires *n:m* inter-ISP trust to exist. The "cost/benefits" of B.1 are *aligned*, because routers upgraded at victim have the greatest fidelity of data to analyze, but offer limited ability to mitigate. However, B.2 and B.3 are *misaligned* since keeping high granularity traffic state may require non-trivial investment from edge ASes, which do not have strong economic incentives to do so for the remote receivers. This reduces the deployability. Finally, our review of the DDoS detection literature reveals a broad lack of discussion of integration with mitigation and enforcement solutions, including where and to whom to communicate the detection results along with the time required to collect, process, and propagate said results.

### C. Mitigation-only Approaches

Upon receiving a DDoS detection alert and, in some cases, traffic classification information, DDoS mitigation approaches are used as a response. Their primary goal is to eliminate or reduce the damage by removing the DDoS traffic as quickly and as early as possible from the network while preserving the quality of service for benign traffic flows. This category does not cover full-suite solutions that combine detection and mitigation; those solutions are discussed in Section V-D.

**C.1 Black-holing** A simple method to stop DDoS traffic is to drop all packets destined for the receiver address at the entry points to the receiver's, or its provider's, network. The use of Remote Triggered Black Hole filtering (RTBH) [101] (see Figure 7) configures routers to announce a black hole route for the receiver's IP address or prefix, dropping all packets that are destined to the receiver. Unfortunately, a significant downside of destination black-holing is the collateral damage

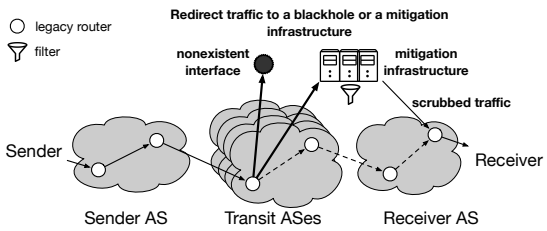


Fig. 7. Traffic redirection to a black-hole or scrubbing center (Market size $2B+)

to legitimate traffic, making it a blunt instrument used only for short periods [102], [103] and when the ISP infrastructure is in peril.

RTBH leverages the standard BGP routing announcements, thus it can be easily configured to mitigate identified DDoS attacks. Further refinements of RTBH enable operators to black-hole only the traffic from specific entry routers to the receiver's AS, or to redirect traffic to a "sinkhole" device for further inspection without requiring router upgrade [101]. Over time the latter has evolved into today's practice of DDoS *Mitigation as a Service* (MaaS, see discussions on traffic scrubbing service below), while the basic RTBH is also expanded with *unicast Reverse Path Forwarding* (uRPF) to drop packets from specific senders [102]–[105]. Similar to destination-based RTBH, source-based RTBH with uRPF drops all traffic to/from the blocked sender IP addresses, not just the traffic to the receiver.

While a readily available solution in terms of no new network implementation, black-holing can incur long response time due to long BGP route propagation delays. A recent study [103] also showed that RTBH could be error-prone due to misconfigurations of BGP policies. Moreover, an RTBH might not be automatically triggered, and even when it is triggered, a high percentage of unwanted traffic remained unmitigated because of non-compliant routers on the path of the attackers [103], [105] or serverless functions in the cloud [106].

To systematize, the "where" of this class is *routers*; the "what" is *none*, it only requires local routing updates; the "who" is *receiver networks* to trigger the BGP announcements and *in-transit/sender networks* to effectuate the black hole route; for the "how," attack traffic is routed to black hole prefixes; the novelty of this approach is that it simply drops victim-bound traffic; for RTBH the "trust coordination" is *n:m*, because $m$ ISPs that are sourcing traffic (to any destination) must be able to trust requests from any of the $n$ ISP networks requesting them to black-hole traffic to their prefix(es); and "cost/benefits alignment" is *aligned* because ISPs provide RTBH service upon victim's request, although it can result in non-trivial collateral damage.

**C.2 Traffic Scrubbing Services** The RFC [101] pointed out, as early as 2004, that when a receiver is under DDoS attack, instead of black holing, one could redirect the victim's traffic to capable devices for further analysis and filtering. Due to rapid growth in DDoS attacks in recent years, and the absence of alternative solutions, traffic scrubbing services followed this

idea and grew to a big MaaS industry [107]–[111]. Upon notification by the victim, the mitigation service provider redirects all the victim-bound traffic to the scrubbing service. This is usually done by changing the victim's DNS record, or through IP anycast [108], or by BGP route update [111]. The scrubbing service uses attack traffic signatures and heuristics, often proprietary solutions, to identify and drop attack packets and forward the rest to the victim (Figure 7). Given today's pervasive use of traffic encryption, scrubbing services need to decrypt the traffic bound to the victim. Cloud-based scrubbing services can be distributed at multiple Points of Presence (PoPs), which help scale up the service and shorten the packet travel paths.

Traffic scrubbing services are readily deployable and can be called on demand because they simply use the existing routing protocols to redirect DDoS victims' traffic to the scrubbing service providers. Scrubbing services are widely deployed in today's Internet as on-demand paid services [109], [110], or by leveraging Content Delivery Networks (CDNs) [112] for those that are already CDN service clients.

To systematize, the "where" is *routers*; the "what" is *none*; the "who" is *local* to MaaS providers only; the "how" is standard BGP route announcement and dedicated infrastructure; the novelty is *no protocol or infrastructure changes*; the "trust coordination" is *1:1*, a business arrangement by victims; and "cost/benefits alignment" is *fully aligned*, but scalability is misaligned (single MaaS provider against distributed attackers).

**C.3 Distributed Traffic Filtering** These types of solutions filter attack traffic using on-path filtering techniques.

**C.3.1 Receiver-controlled Traffic Filtering by Routers** This class of solutions enables receivers, or their ISPs, to install traffic filters at network routers in upstream ASes, so that attack traffic can be distributively dropped at the edge. First, AITF [113] proposes to let receivers push filters "deep" into the Internet, close to all potential attack traffic senders (Figure 8). Whenever a DDoS is detected, the receiver generates filter rules based on the signature of attacking traffic obtained from a separate detection solution deployed in the AS (❷), and then propagates the filter rules to upstream border routers. To identify the border routers along the traffic path, AITF requires participating routers to put their IP addresses on each packet being forwarded (❶). That is, AITF uses packets to carry *extra state* of traffic paths. The AS border routers of the receiver first start filtering and further set up a three-way handshake with the next upstream border router towards the attacker(s). This process is repeated until the AS border routers of the attack sender install these filters (❸). This solution lets border routers along different attacking paths drop packets, creating a distributed mitigation against DDoS attacks.

A later work, TRACK [114], also makes use of packet marking to install filters at remote routers but differs from AITF in the marking and traceback mechanisms. Instead of marking packets with router IP addresses, TRACK lets routers mark packets probabilistically concatenating the incoming interface's port number of each path router as a unique identifier. StopIt [115] proposes to prevent forged signals or marks. It uses overlay servers for secure exchanges of packet filtering messages among ASes. StopIt further requires prior key negotiation among ASes to secure message authentications.

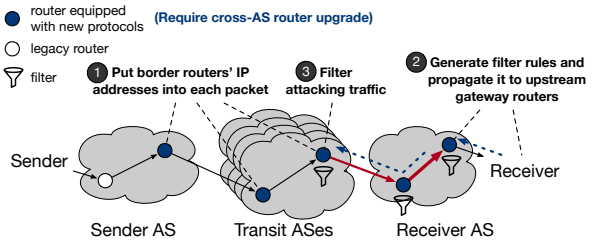


Fig. 8. Receiver propagates filters to upstream routers in AITF [113] (Cited by 250+)

To achieve distributed mitigation, the shared requirements among the router-based filtering solutions are secure cross-AS coordination and potential router modifications to perform packet filtering. These requirements share similar concerns with most previously discussed solutions regarding the costs and incentives of remote participating parties.

Systematizing, the "where" is *routers*; the "what" is *traffic filters*; the "who" is *on-path* access provider ISPs (and optionally upward); the "how" is *new peering procedures*, new negotiation/authorization, then destinations push filters; the novelty of this approach is *reduced data-plane state*, but comes with increased control-plane complexity and state; the "trust coordination" is *1:n*, inter-ISP trust is needed; and "cost/benefits alignment" is *partially aligned*, access providers get paid to provide service, but *not all* customers may want/benefit from the upgrade and there is a *potential negative impact* to other customers.

**C.3.2 Filtering using Overlays** To avoid making changes to deployed routers, some researchers proposed the use of overlays as a distributed firewall to filter attacking traffic. DDoS defenses using overlays include SOS [116] and Mayday [117]. Specifically, senders' traffic travels through the overlay to reach a receiver, and the receiver can push packet filtering rules throughout the DDoS mitigation overlay to filter out unwanted traffic. The strength of overlay-based defenses depends on the number of overlay servers, their locations in the network, and the aggregate traffic volume they can sustain. A mitigation service provider must deploy a sufficient number of nodes in the overlay network to match the level of protection that it plans to offer. Similar to the deployment of CDNs, overlay-based DDoS defenses make use of DNS to redirect all the traffic to the overlay. This, therefore, does not require any change to the underlying network infrastructure.

Systematizing, the "where" is *endhosts* and *overlay-routers*; the "what" is *endhosts*; the "who" is *endpoints* and *overlay infrastructure*; the "how" is all transmitted/received service traffic uses overlay infrastructure; the novelty is *no changes* needed to existing routing infrastructure; the "trust coordination" is a *1:1*, business relationship; and "cost/benefits alignment" is *fully aligned*, but *scalability is mismatched* because the overlay must be provisioned to support the full service (and any unmitigated attack traffic) load.

**C.4 Moving Target Defenses (MTD)** These solutions change target locations dynamically, to challenge attackers.

**C.4.1 Address Changing** When a DDoS attack is detected, this type of solution avoids the receivers being attacked by changing their IP addresses continuously [118]–[124]. Although these solutions are readily deployable with the existing network infrastructure, they usually rely on a layer of indirection between senders and the receiver, for example, by leveraging a number of cloud-based proxy servers to control the routing of the senders' packets to the receiver. Thus through dynamic packet routing, moving target approaches can extend the DDoS defenses beyond web services already supported by commercial CDN overlays. Different from overlay DDoS defenses, moving target solutions do not filter traffic but swiftly alter the location of the target and inform legitimate users of the new location while attack traffic persists on the old location. This is achieved through the use of client puzzles (see A.4 in Section V-A) that contain the new location of the target. This class of defenses assumes that unsophisticated attackers would be unable to solve the client puzzles to "follow" the receiver when it hops between different IP addresses and that it would be difficult for a botmaster to coordinate a large network of bots with diverse resources and capability to solve puzzles.

Systematizing, the "where" is *endhosts* and *overlay-routers*; the "what" is *endhosts* and *service-routers*; the "who" is *endpoints* and *overlay* (local); the "how" application-layer puzzles to transmit to "moving" service location; the novelty is no changes needed to existing routing infrastructure; the "trust coordination" is *1:1*, business incentivized; and "cost/benefits alignment" is *fully aligned*, but endpoints must use MTD admission and additional IP destinations are needed to move the service between.

**C.4.2 Anycast Load Shifting Based Defenses** This type of approach protects a receiver by IP anycast and mechanisms to shift DDoS traffic among multiple anycast sites of the receiver. IP Anycast deployments localize the impact of DDoS attacks by replicating the services at multiple locations that are split into separate catchments in which traffic is redirected to the closest anycast site by the inter-domain routing system [125]. When a volumetric DDoS attack overwhelms a particular anycast site, some portion of the attack traffic is automatically redirected to other anycast sites via BGP route announcements [126]. Anycast based mitigation techniques have been used in commercial solutions such as AT&T [127] and Akamai [128]. A recent work proposed by Rizvi *et al.* automatically generates a BGP response playbook for an anycast deployment [129]. Such a playbook will list rules that can influence the existing catchments with traffic engineering technique such as AS path prepending and AS poisoning. Upon receiving a DDoS alert, the system can estimate the total attack traffic load by calculating the loss rate of the normal load of each site based on the non-attack period statistics. Based on the attack load, the system or operator can pick one or more rules to balance the total load among the available anycast sites for mitigation.

Compared to the address changing approaches, in this mitigation system, the attacker always has the target IP address but the dynamic load shifting forces the attacker to either spend more to overwhelm all the anycast sites at the same time or change the attack senders regularly to target a particular site for a limited time (BGP route convergence delay).

Systematizing, the "where" is *routers*; the "what" is *none* (no changes needed); the "who" is *receiver networks* (local sources only); the "how" is standard BGP anycast; the novelty is that it requires no protocol or infrastructure changes; the "trust coordination" is *none*; and "cost/benefits alignment" is *fully aligned*, but *scalability is misaligned* (single anycast service owner against distributed attackers).

**Summary** The "where" of solutions in this category each fall into one of two buckets: routers (C.1, C.2, C.3.1, and C.4.2) or endhosts and overlay routers (C.3.2 and C.4.1). For the solutions at routers, the "what" and "how" consist of three major components: First, while the various solutions differ in where and how to get the information to classify traffic, all solutions use one of the two ways to obtain traffic path information, either by letting packets carry the path information [89], [113], [114], [130] or letting routers remember the trace information [87], [88]. Second, there is a need to trace back the attack traffic, so that packet filters can be installed close to attack senders. Third, the packet filters have to be installed close to attack senders to be effective. This can be done either by hop-by-hop signaling (*e.g.*, AITF, TRACK), or by building an overlay network (*e.g.*, StopIt); the latter also adds crypto protection to the filter exchanges. The "who" for C.1, C.2, and C.3.1 are senders, mitigation providers, and access/transit ISPs while for C.4.2 the "who" is receiver network only. Solutions in C.3.2 use their overlay infrastructures, and C.4.1 solutions use L7 puzzles to transmit service locations.

Our systematization identifies that the cost/benefits alignment of solutions in this category are all either aligned or fully aligned. Yet, even though the rich literature in DDoS mitigation provides a plethora of solutions, the majority of these approaches have not been deployed in practice. The few solutions being adopted today include simple black-holing (*e.g.*, RTBH [102]), mitigation-as-a-service overlays (*e.g.*, SOS [116], Mayday [117], traffic scrubbing technology [107]–[111]) and anycast based mitigation systems [127]. Perhaps our most fundamental insights in this category come here: the most widely deployed class of solutions that we surveyed are those solutions that have aligned costs with benefits. Further, those solutions that reached successful deployment uptake had the "novel" systematization design pattern of not requiring changes to the inter-domain routing infrastructure. Said simply, they require *no changes to the deployed infrastructure*, these solutions can be, and have been, deployed quickly even though they merely treat the symptoms.

### *D. Holistic Approaches*

These approaches integrate detection and mitigation coherently and systematically. Here, again, we provide the systematization in the summary section.

**D.1 SDN/NFV based Detection and Mitigation** Many proposals explore the use of SDN for DDoS detection and

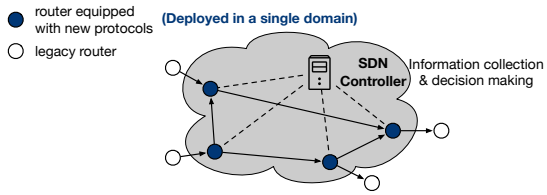


Fig. 9. DDoS detection and mitigation with SDN.

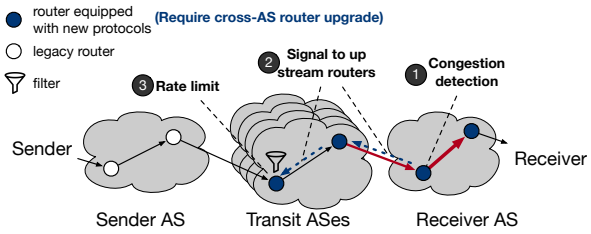


Fig. 10. ACC [57] (Cited by 1090+) lets routers keep per-flow state to detect congestion.

mitigation [49], [68], [131]–[142]. SDN provides a single administrative platform within an organization, enabling quick information collection for attack detection, and updates of network configurations (*e.g.*, blackhole, redirection) for mitigation (Figure 9).

NICE [132] utilizes OpenFlow's APIs to monitor traffic and apply graph-based analytical models with a constant threshold to detect attack events. When a DDoS-like event is detected, NICE utilizes the network controller to change the flow table on each switch/router, redirecting attacking traffic to cloud scrubbing centers or to hardware appliances for Deep Packet Inspection (DPI). SnortFlow [133] performs DPI via Snort-based Intrusion Prevention System (IPS) coupled with SDN. DPI techniques use packet and flow information to detect and filter-out attack traffic but are expensive and cannot be applied to encrypted traffic (some customers might share their crypto keys with mitigation providers for decryption).

ProDefense [49] utilizes an exponentially weighted moving average based threshold for real-time attack detection and makes the detection filter adaptive to meet application-specific requirements. When an attack is detected, ProDefense can drop packets following predefined rules, blackhole certain ports, or redirect traffic to a DPI node for further filtering. Recently the use of switch-native approaches for volumetric DDoS defense that can run detection and mitigation functions entirely inline on switches was proposed [131]. Similarly, in [143] the authors leverage Network Function Virtualization (NFV) to allow a flexible capacity and functionality control of SDN programmable switches to quickly deploy DDoS defenses. The real-time mitigation performance of these approaches is limited by the current TCAM capacity related scalability issues thus reducing their deployability. Jung et al. recently proposed a novel in-switch ACL system that can perform 168x faster than state of the art to address this issue [144].

Deploying SDN-based DoS solutions is straightforward for networks that fully support SDN devices. However, when it comes to the distributed inter-domain nature of DDoS, SDN's intra-domain scope faces challenges. First, the centralized control plane may well become a target of focused attacks given its critical importance [41], [145]–[147]. Thus additional mechanisms are required to prevent, detect, and mitigate DDoS attacks towards the controllers [145], [148]–[154]. Second, because SDN deployments are confined within individual administrative domains, this scope limitation runs counter to the inter-domain nature of volumetric DDoS attacks. When a large-scale DDoS attack occurs, its traffic volume, *e.g.*, what was seen in the Mirai botnet attack [53], [155], can overwhelm SDN defenses that are not deployed across different administrative domains to match the distributed nature of the threat.

**D.2 Congestion Control with Enforcement by Routers** Treating DDoS flooding attacks as a special case of network congestion, several works aim to add router-enforced congestion detection and control as DDoS mitigation [57], [91], [156]. Different from TCP congestion control and other transport protocols for hosts, these proposed congestion control approaches make routers across ASes drop excessive packets. However, different from other holistic approaches discussed in this section, congestion control approaches do not distinguish attack traffic from legit traffic; instead, they treat congestion.

ACC [57]/Pushback [156] requires routers to *keep additional states* of the loss rate of each underlying traffic flow (defined by destination IP address). When it reaches a threshold determined by the policy or historical loss rate (❶), a router can signal its upstream routers (❷) to perform rate-limiting (❸). Instead of keeping states at routers, NetFence [91] lets packets carry congestion marks to receivers to trigger the traffic reduction. Whenever an excessive load is detected by a router, the router generates a congestion mark with a secret symmetric key and adds it to packets to receivers. The receivers can send responses piggybacking the mark back to the senders. When the signal is seen by trusted border routers connected to corresponding sender networks, it can rate-limit the sender. NetFence requires global scale deployment at all routers and may involve negotiation among ASes.

**D.3 Collaborative Detection & Mitigation** In this section, we investigate router overlay based and high-level information exchanged based collaborative approaches.

**D.3.1 Router Overlay Based** Router overlay based approaches [100], [157] promote solutions where routers from the same AS or across ASes exchange information, evaluate emerging risks, and defend against DDoS. DefCOM [157] establishes an overlay to coordinate DDoS alerts and responses leveraging existing detection and mitigation mechanisms. DefCOM defines traffic policing rules to indicate traffic risk level and the message formats for peer-to-peer communication for status synchronization. Upon attack detection, the classifiers will mark traffic with different risk levels, and rate limiters will apply different limits based on the stamps. In addition, FireCol [100] proposes that the ISPs should coordinate for DDoS detection and defense using an overlay built on routers around the victim. Traffic with a high-risk score triggers communication across routers of the same distance from the receiver, enabling these ISPs to block attack-related IP sources

to mitigate the DDoS attack. FireCol requires router upgrade and historical traffic state in the overlay nodes but offers a service-like subscription to motivate deployment.

**D.3.2 High-Level Information Exchange** This class of approaches aim to achieve collaborative DDoS defense between different parties by acting as a high-level communication channel without proposing any change to the routers. For example, IETF's signaling architecture (DOTS) allows networks under DDoS attack to request help from their upstream network or a remote third party mitigation service provider regardless of the specific detection or mitigation system [58]. In DOTS, networks that demand mitigation deploy DOTS clients and the networks that provide mitigation deploy DOTS servers. The DOTS client and server create TLS based data and signal channels. The data channel is used for transferring initial configurations such as ACL rules before the attack and the signal channel is used for sending mitigation requests, receiving mitigation status and updating the ACLs during an attack. In the case of a DDoS alert, the DOTS client signals the DOTS server for mitigation. The server might handle this request via its local mitigation system or conveys it recursively to a third-party mitigator via another DOTS signal. During a mitigation, the DOTS server and client regularly share DDoS related telemetry with each other to refine the mitigation actions. Another recent work, DXP [158], allows multiple ISPs to share DDoS telemetry such as reflector server IPs, victim IPs and attack traffic volume with each other via a publisher/subscriber system to achieve better level of detection and mitigation. Rodrigues *et al.* [159] proposed to use the blockchain systems as a distributed immutable database to *signal and share* the DDoS detection results, and mitigation information. This system utilizes smart contracts to share whitelisted or blacklisted IP addresses among peers, who can be AS operators.

**Systematization and Summary** Holistic solutions aim to offer complete and comprehensive mitigation to DDoS attacks either by performing both detection and mitigation, or by providing a communication fabric to coordinate existing defenses. While their completeness is appealing, holistic solutions appear to have received limited academic focus. The "what" of all of the solutions except D.3.2 in this categorization require *router upgrades*, however the "who" varies from *individual networks* (D.1) who need *no* additional "trust/coordination" with other administrative domains, to the need for *global deployment* (D.2) which needs *inter-ISP* "coordination/trust", to deployment targets in *access provider ISPs and upward towards transit providers* (D.3.1) where an *n:m* "trust/coordination" model is needed. For D.3.2, only parties willing to collaborate deploy the solution thus *1:n* "trust/coordination" is required.

Once again, perhaps the most telling aspect of our systematization is the "cost/benefits" alignment. Although solutions in D.1 *align* costs with benefits, these solutions are inherently limited in scope, and hence in their effectiveness. They propose intra-domain solutions to counter inter-domain DDoS attacks with the global scale nature. In contrast, D.2 solutions *misalign* "costs/benefits," as deploying routers do not gain benefits. They also incur the risk of false positives leading to inter-domain packet drops, a central risk and aversion to transit operators. Finally, solutions from D.3 present an idealized goal of all Internet operational parties joining forces together for the common good, and cooperating in real time to mitigate DDoS. Unfortunately, these solutions not only leave open the question of how participating parties are compensated for the cost, but also raise another challenging problem of how to establish and maintain inter-administrative trust and coordination in order to effectuate these solutions. Without clarifying either the cost/benefits or trust relationships, these deployment will likely not succeed at any scale.

## VI. Distilling DDoS Mitigation Design Patterns

In the course of systematizing the voluminous corpus of the DDoS literature, several important points emerge: many independently proposed solutions advocate repeated design patterns, and those proposals that faced deployment challenges have important commonalities, as do those proposals that have been successfully deployed.

### A. Repeated Design Patterns

From our description in Section IV, the vast majority of DDoS defense solutions share a few design approaches. First, they take a distributed approach: this includes all of the preventive, all of the detection-only solutions, and almost all of the holistic solutions. Such distributed solutions, in principle, seem to head in the right direction: given that DDoS attacks are distributed, effective solutions should be distributed as well, to detect and block attack traffic near sources before it aggregates.

Second, and related, these distributed solutions all require real-time, secure coordination among all participating parties. In today's operational Internet, however, a standard approach to coordination across multiple parties does not exist. When coordination is necessary, it is carried out by operators manually, we surmise because of *IP property 6* as defined above in Section II-A.

Third, since stateless IP routers can only forward all packets indiscriminately, mitigating DDoS requires additional necessary information to sort out "allowed" from "disallowed" traffic. These distributed solutions all propose to add such information in the packet forwarding process via one of two means, either installing the information at routers, or carrying it on the packets. Either way, they require changes to the deployed Internet infrastructure, which correlates to *IP property 5* from Section II-A.

Among the mitigation-only approaches, category C3 solutions (distributed traffic filtering) direct traffic to a distributed overlay upon receivers requests, and sort out allowed from disallowed traffic on the overlay. Thus, this category shares the first two design patterns but avoids the third one.

### B. Solutions Facing Deployment Challenges

The DDoS defenses that share the above mentioned three repeated design patterns face a few insurmountable issues when being applied to the operational Internet.

*First*, distributed DDoS defense solutions require *automated, secure, multiparty* coordination, a function that is far beyond IP's basic properties as described in Section II-A At the moment, the only global coordination across all the operators is policy-controlled BGP routing exchanges.

*Second*, such coordinated efforts would succeed only if the benefits for each of all the involved parties were to be well understood. The benefit from the coordinated global routing system meets this requirement: everyone depends on the reachability provided by BGP to support its local users.

*Third*, modifying the deployed infrastructure demands financial and operational investments. Thus, effective solutions must align the costs and expected gains derived from the required infrastructure modifications. As we show repeatedly in this paper, deployment traction suffers when the parties shouldering the costs are not provided with corresponding economical benefits and incentives to recoup their investment.

### C. Deployed Solutions and Lessons Learned

Examining successfully deployed approaches, we find solutions that do not require changes to the deployed router infrastructure, do not require multiparty coordination by being *single-party* provided solutions, and they are mitigation-only solutions. This is because prevention and detection require distributed approaches to be most effective, which in turn requires multiparty coordination. We observe that these solutions are not represented in the literature, but are solely found in operational deployments.

These mitigation-only solutions can be roughly divided into two classes: application layer and network/transport defenses, which are described in Section V-C. Examples of the former are proxies provided by web hosting (*e.g.*, Cloudflare [160], Radware, Neustar) and CDN providers (*e.g.*, Akamai, Fastly). CDNs have traditionally used DNS to redirect traffic bound to their paid clients to CDN servers and have aimed to scale Web content distribution. More recently, many have evolved to integrate CDN services with DDoS mitigation. These are application-layer services with associated costs.

Different from the above always-on solutions, network/transport defenses can be triggered upon DDoS detection via *external* means. This category of solutions includes Black Holing and Mitigation as a service (MaaS). The former affects all traffic, thus it is generally handled manually by network operators to minimize the significant collateral damage associated with its use. The latter can be activated dynamically, using either DNS redirect or BGP routing announcements to haul DDoS victims' traffic to scrubbing centers [1] to sort out good versus bad traffic. Conceptually, this class of solutions appears straightforward[1], has a unique advantage of being quick to deploy, and thus provides DDoS mitigation on-demand as a shared service. Scrubbing centers are relatively centralized compared to the extremely distributed nature of attackers. Not only must they possess high processing capacity, but hauling high volumes of attack traffic to those centers requires high network capacity and can also cause collateral damage.

Although these deployed DDoS mitigation systems work can effectively today, there exists a clear economical imbalance between DDoS offenders and defenders. While DDoS defense services must pay to provision capacity, attackers can launch attacks at will with essentially zero cost. There exist a vast number of vulnerable devices that (once compromised) are free to use, and the number of such devices is ever-increasing, a characteristic of *IP properties 1-3*, as described in Section II-A. Compounded by ever-increasing network bandwidth, the volume of DDoS attacks has been growing exponentially with time. This trend forces the DDoS defenses to invest ever-increasing resources in a tail-chasing spin, and results in only a small number of providers being able to shoulder the weight of provisioning ever-growing capacity, contributing to a more centralized and privatized Internet defense force.

### D. Where the Future Direction Lies

Our analysis of the literature and our limited empirical knowledge of today's deployed DDoS mitigation solutions lead us to the following observations. First, it is infeasible to develop DDoS mitigation solutions based on the assumption that today's deployed IP network infrastructure can be easily or incrementally changed. Consequently, today's deployed solutions can only work on top of the existing network infrastructure, *e.g.* using overlays such as CDNs and MaaS[2] – both of these sit between end hosts and DDoS victims to filter out bad packets. Second, it is also infeasible to assume a quick rollout of *secure* multiparty coordination on today's TCP/IP Internet, without first addressing the economic incentive question and building a multiparty security framework, a result of *IP property 6* from Section II-A. All today's deployed solutions are provided by single-parties and for paying customers only, which do not require multiparty coordination with properly aligned costs and benefits. Third, given the Internet itself is a vastly distributed system, it is questionable, in the long run, whether it is viable to rely on a small group of for-profit companies to build capacities to match global DDoS' ever increasing volume [161].

The above observations suggest that a promising direction for DDoS mitigation should be *architecting* an overlay solution. In contrast to the existing scrubbing centers and CDN-based mitigations which are proprietary solutions operated by *individual* companies, a new overlay architecture could foster *open interconnections* of all Internet parties/participants. Scrubbing centers and CDNs serve only those who can afford to pay, and are brute-force solutions that come with a questionable economical future as we mentioned earlier. However, an overlay approach shares the same spirit as IP started with. IP was originally a means of interconnecting all the computers by building an overlay on top of the existing telephone infrastructure.

If the community is open to exploring the direction of architecting a new overlay, we may wish to take into consideration solutions that not only facilitate DDoS defenses, but also

[1]In reality, examples vary in form and function, can have approaches with very nuanced differences as differentiators by different MaaS providers.

[2]Although packet scrubbing centers have far less servers than CDNs, they could be viewed as specialized overlays interconnected via tunnels.

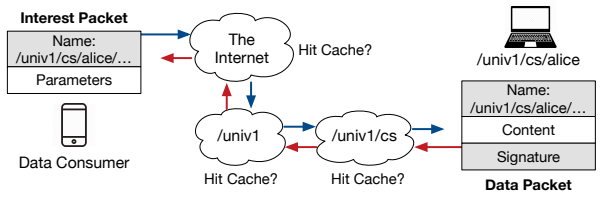


Fig. 11. Named secured data follows the path laid by its interest packet back to the requester

address *other* architectural needs and persistent problems that have faced Internet for years. These include network security, multicast delivery, scalable data dissemination, delay-tolerant networking, mobile ad hoc networking (MANETs), among others. At the moment, proposals in each of these problem areas develop their own solutions *in isolation from the others*. Could a new architectural design incorporate them all into a coherent architecture with innately aligned costs/benefits?

Indeed, some new overlay architectures have been developed in recent years. One example is SCION [162], which addresses two of the *IP properties* from Section II-A. Defining secured and highly available point-to-point connectivity as its building block for networking, SCION builds an overlay to secure the Internet's *routing plane*. It integrates the promising ideas in improving Internet routing security and availability that have been accumulated over time. This includes (i) establishing multiple trust roots, one for each *isolation domain* (ISD) which is made of a collection of autonomous systems, (ii) securing all routing information exchange, and (iii) enabling end hosts to control data paths via source routing, and thereby accommodating *IP property 5*. By building an overlay on top of the existing Internet routing infrastructure, SCION does not require changes to the existing routers, although it still needs to address the challenge of establishing secure coordination among multiple ISDs on the overlay, which is done out-of-band, leaving *IP property 6* unaddressed. As an overlay for Internet routing, SCION uses bandwidth reservations to prevent DDoS attacks to address *IP property 1*. However, given the overlay would cover no more than a small portion of the Internet in its initial rollout, it is unclear how effective such prevention solution could be. For example, if nascent deployment of SCION provisions a SCION router at a victim AS and one at a source AS (as an overlay, connected by underlying transit infrastructure), then the reservation at the victim would not be able to help because the upstream transit routes would already be congested during a DDoS.

Another architectural overlay design is Named Data Networking (NDN) [163], [164], which addresses all six *IP properties* from Section II-A. Different from SCION, NDN aims to build a secure foundation that can better serve today's widely diverse Internet applications. NDN's basic building block is semantically named and secured data objects. With a data-centric design, NDN secures data directly and utilizes a stateful forwarding plane [165], which makes each requested data packet follow the path laid by the corresponding interest packet to go back to the requester, architecturally addressing *IP properties 4 & 5*. Additionally, NDN mandates that all data be signed by producers' keys, so that *all* data in NDN carries explicit origin authenticity, thereby addressing *IP property 3*. Figure 11 shows an example of a data consumer retrieving a piece of named secured data over NDN. Further, NDN uses semantic names in forwarding, making its namespace unbounded, thereby addressing *IP property 2*. NDN also has innate support for multicast delivery, ubiquitous in-network caching, and suitability to delay-tolerant network (DTN) and MANET environments. Furthermore, NDN's built-in stateful forwarding plane can be directly used to mount DDoS mitigation solutions, which addresses *IP property 1*. Indeed, a number of such solutions have been proposed on using NDN's stateful forwarding plane to build a fully effective DDoS mitigation strategy [166]–[175]. Given that DDoS is a fundamentally data-plane resource exhaustion problem, these proposed solutions share a common approach of utilizing NDN's unique stateful data-plane to detect abnormal behavior in traffic and to push back offensive traffic to their originating points, but differ in the specifics in the detection and mitigation designs.

Regarding deployment hurdles, NDN avoids the need of changing the existing router infrastructure by being an overlay, which also addresses *IP property 6*. But fundamentally different from SCION, NDN offers incentives for applications to be developed over NDN to benefit from its built-in security support and resilient, scalable data delivery. This application incentive is well suited to lead to an application-driven deployment rollout, resulting in an *"edge-in"* rollout model (which has also been coined "limited domains" in the literature [176], [177]). Furthermore, since NDN's capabilities do not require any specific network path configuration, it can get the ball rolling by tunneling through the Internet infrastructure to connect NDN-deployed islands, or operate through any portion of the infrastructure that has not deployed NDN transport natively. NDN's built-in security support also natively enables secure multiparty coordination. That is, there is no requirement for coordination with parties that have not deployed NDN yet.

## VII. Conclusion

DDoS poses a perennial threat to the very fabric of the Internet, and has received significant research and development attention from both academics and industry practitioners. Billions of dollars have been invested in defenses, but still, even more billions of dollars have been lost due to DDoS-induced outages. In this paper, we set out to understand the underlying challenges in solving the DDoS problem and provide insights into the deployment obstacles. When taking a broad view of the DDoS-related literature, as we have done in this work, many important lessons become apparent. Among these are the clear need for a solution to the DDoS plague, but also an implicit caution that infrastructure changes cannot be sought lightly. The aggregate picture that emerges reveals that the majority of DDoS remediations, which appear attractive but stay at the "being proposed" stage, require network-wide infrastructure upgrades, and that effectively mitigating DDoS while creating aligned economic incentives at the same

time is a grand challenge. The novel systematization we developed and use throughout this paper is summarized in an online table available at [59], which makes it qualitatively clear that most solutions require modifications to the existing routing infrastructure but cost/benefit misalignment limits their deployability. We posit that architectural changes are indicated by the extreme threat perennially posed by DDoS, but that when changes to the architecture are made, there will be a critical *opportunity* for those changes to fulfill not just the architectural promises and requirements of DDoS remediation, but *also* for the evolving *the broader needs* of the Internet. As illustration, architectural overlays are a repeated design pattern throughout the DDoS remediation literature. However, when deploying infrastructure changes to adopt such an architecture, that solution should *also* have utility for other popular Internet applications too (not just DDoS). With such a proposition, the extensibility of Named-Data Networking (NDN) to perform overlay-style DDoS protection while also innately meeting other Internet application requirements paves an important *strategic* path forward. NDN is an incrementally deployable architecture whose fundamental nature befits both the diverse needs of Internet applications *and* foundationally resists DDoS, innately. The broader DDoS mitigation community needs to invest in a properly aligned architectural solution in order to finally win the war against DDoS.

## REFERENCES


[1] E. Osterweil, A. Stavrou, and L. Zhang, "21 years of distributed denial-of-service: A call to action," *Computer*, vol. 53, no. 8, pp. 94–99, 2020.

[2] MarketWatch Inc. (2021) Ddos protection market insights by size, share, future growth and forecast from 2021-2025. [Online]. Available: https://www.marketwatch.com/press-release/ddos-protection-market-insights-by-size-share-future-growth-and-forecast-from-2021-2025-2021-03-31

[3] Market Data Forcase Inc. (2021) Global ddos protection and mitigation market research. [Online]. Available: https://www.marketdataforecast.com/market-reports/ddos-protection-mitigation-market

[4] J. Grady, C. Christiansen, C. Price, and C. Richmond, "Worldwide ddos prevention products and services 2013-3017 forecast," *IDC*, 2021.

[5] S. Kar. (2021) Looking back at ddos in 2014: Most enduring attack trends. [Online]. Available: https://siliconangle.com/2015/01/05/looking-back-at-ddos-in-2014-most-enduring-attack-trends/

[6] Cloudflare Inc. (2021) Famous ddos attacks — the largest ddos attacks of all time. [Online]. Available: https://www.cloudflare.com/learning/ddos/famous-ddos-attacks/

[7] P. Nicholson. (2021) Five most famous ddos attacks and then some. [Online]. Available: https://www.a10networks.com/blog/5-most-famous-ddos-attacks/

[8] Cision US Inc. (2021) 20 years of ddos attacks: Real world cyber reflections. [Online]. Available: https://mb.cision.com/Public/13800/2089427/b3cd773059da6454.pdf

[9] J. van Aalst, "Using google scholar to estimate the impact of journal articles in education," *Educational researcher*, vol. 39, no. 5, pp. 387–400, 2010.

[10] C. Neuhaus, E. Neuhaus, A. Asher, and C. Wrede, "The depth and breadth of google scholar: An empirical study," *portal: Libraries and the Academy*, vol. 6, no. 2, pp. 127–141, 2006.

[11] The DBLP team. (2019) Dblp: computer science bibliography. Online; Available at https://dblp.org/. [Online]. Available: https://dblp.org/

[12] Z. Durumeric, E. Wustrow, and J. A. Halderman, "Zmap: Fast internet-wide scanning and its security applications," in *22nd {USENIX} Security Symposium ({USENIX} Security 13)*, 2013, pp. 605–620.

[13] M. Mahmoud, M. Nir, A. Matrawy, *et al.*, "A survey on botnet architectures, detection and defences." *Int. J. Netw. Secur.*, vol. 17, no. 3, pp. 264–281, 2015.

[14] F. Ali, "Ip spoofing," *The Internet Protocol Journal*, vol. 10, no. 4, pp. 1–9, 2007.

[15] C. Rossow, "Amplification hell: Revisiting network protocols for ddos abuse." in *NDSS*, 2014.

[16] K. Bock, A. Alaraj, Y. Fax, K. Hurley, E. Wustrow, and D. Levin, "Weaponizing middleboxes for {TCP} reflected amplification," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 3345–3361.

[17] J. D. Howard and T. A. Longstaff, "A common language for computer security incidents," Sandia National Labs., Albuquerque, NM (US); Sandia National Labs, Tech. Rep., 1998.

[18] J. Howard, "An analysis of security incidents on the internet," *PhD thesis, Carnegie Mellon University*, 1998.

[19] F. Kargl, J. Maier, and M. Weber, "Protecting web servers from distributed denial of service attacks," in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 514–524.

[20] R. K. C. Chang, "Defending against flooding-based distributed denial-of-service attacks: a tutorial," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 42–51, 2002.

[21] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *IEEE computer*, vol. 35, no. 10, pp. 54–62, 2002.

[22] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: Real vulnerabilities and practical solutions." in *USENIX security symposium*, vol. 12. Washington DC, 2003, pp. 2–2.

[23] A. Hussain, J. Heidemann, and C. Papadopoulos, "A framework for classifying denial of service attacks," in *Conference on Applications, technologies, architectures, and protocols for computer communications*, 2003, pp. 99–110.

[24] J. Mirkovic and P. Reiher, "A taxonomy of ddos attack and ddos defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–53, 2004.

[25] C. Douligeris and A. Mitrokotsa, "Ddos attacks and defense mechanisms: classification and state-of-the-art," *Computer Networks*, vol. 44, no. 5, pp. 643–666, 2004.

[26] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, "Denial-of-service attack-detection techniques," *IEEE Internet computing*, vol. 10, no. 1, pp. 82–89, 2006.

[27] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *ACM Computing Surveys*, vol. 39, no. 1, pp. 3–42, 2007.

[28] D. R. Raymond and S. F. Midkiff, "Denial-of-service in wireless sensor networks: Attacks and defenses," *Pervasive Computing*, vol. 7, no. 1, pp. 74–81, 2008.

[29] G. Badishi, A. Herzberg, I. Keidar, O. Romanov, and A. Yachin, "An empirical study of denial of service mitigation techniques," in *Symposium on Reliable Distributed Systems*, 2008, pp. 115–124.

[30] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection," in *Third International Conference on Emerging Security Information, Systems and Technologies*, 2009, pp. 268–273.

[31] N. Ahlawat and C. Sharma, "Classification and prevention of distributed denial of service attacks," *International Journal of Advanced Engineering Sciences and Technologies*, vol. 3, no. 1, pp. 052–060, 2011.

[32] Z. Chao-yang, "Dos attack analysis and study of new measures to prevent," in *International Conference on Intelligence Science and Information Engineering*, 2011, pp. 426–429.

[33] K. W. Ghazali and R. Hassan, "Flooding distributed denial of service attacks-a review," *Journal of Computer Science*, vol. 7, no. 8, p. 1218, 2011.

[34] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of service attacks in wireless networks: The case of jammers," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 245–257, 2011.

[35] A. Mishra, B. B. Gupta, and R. C. Joshi, "A comparative study of distributed denial of service attacks, intrusion tolerance and mitigation techniques," in *European Intelligence and Security Informatics Conference*, 2011, pp. 286–289.

[36] L. Zhang, S. Yu, D. Wu, and P. Watters, "A survey on latest botnet attack and defense," in *International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 2011, pp. 53–60.

[37] M. Aamir and M. A. Zaidi, "A survey on ddos attack and defense strategies: from traditional schemes to current techniques," *Interdisciplinary Information Sciences*, vol. 19, no. 2, pp. 173–200, 2013.

[38] B. Harris, E. Konikoff, and P. Petersen, "Breaking the ddos attack chain," *Institute for Software Research*, 2013.

[39] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.

[40] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (sdn) and distributed denial of service (ddos) attacks in cloud computing environments: A survey, some research issues, and challenges," *Communications surveys & tutorials*, vol. 18, no. 1, pp. 602–622, 2015.

[41] Q. Yan and F. R. Yu, "Distributed denial of service attacks in software-defined networking with cloud computing," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 52–59, 2015.

[42] M. Masdari and M. Jalali, "A survey and taxonomy of dos attacks in cloud computing," *Security and Communication Networks*, vol. 9, no. 16, pp. 3724–3751, 2016.

[43] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, "Distributed denial of service (ddos) resilience in cloud: review and conceptual cloud ddos mitigation framework," *Journal of Network and Computer Applications*, vol. 67, pp. 147–165, 2016.

[44] S. Behal and K. Kumar, "Trends in validation of ddos research," *Procedia Computer Science*, vol. 85, pp. 7–15, 2016.

[45] B. Gupta and O. P. Badve, "Taxonomy of dos and ddos attacks and desirable defense mechanism in a cloud computing environment," *Neural Computing and Applications*, vol. 28, no. 12, pp. 3655–3682, 2017.

[46] K. Kalkan, G. Gur, and F. Alagoz, "Defense mechanisms against ddos attacks in sdn environment," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 175–179, 2017.

[47] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "Ddos attacks in cloud computing: Issues, taxonomy, and future directions," *Computer Communications*, vol. 107, pp. 30–48, 2017.

[48] S. Behal, K. Kumar, and M. Sachdeva, "Characterizing ddos attacks and flash events: Review, research gaps and future directions," *Computer Science Review*, vol. 25, pp. 101–114, 2017.

[49] N. Z. Bawany, J. A. Shamsi, and K. Salah, "Ddos attack detection and mitigation using sdn: methods, practices, and solutions," *Arabian Journal for Science and Engineering*, vol. 42, no. 2, pp. 425–441, 2017.

[50] N. S. Rao, K. C. Sekharaiah, and A. A. Rao, "A survey of distributed denial-of-service (ddos) defense techniques in isp domains," in *Innovations in Computer Science and Engineering*. Springer, 2019, pp. 221–230.

[51] B. A. Khalaf, S. A. Mostafa, A. Mustapha, M. A. Mohammed, and W. M. Abduallah, "Comprehensive review of artificial intelligence and statistical approaches in distributed denial of service attack and defense methods," *IEEE Access*, vol. 7, pp. 51 691–51 713, 2019.

[52] H. Griffioen, K. Oosthoek, P. van der Knaap, and C. Doerr, "Scan, test, execute: Adversarial tactics in amplification ddos attacks," in *ACM CCS*, 2021, pp. 940–954.

[53] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, *et al.*, "Understanding the mirai botnet," in *USENIX security symposium*, 2017, pp. 1093–1110.

[54] N. Agrawal and S. Tapaswi, "Defense mechanisms against ddos attacks in a cloud computing environment: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3769–3795, 2019.

[55] A. Praseed and P. S. Thilagam, "Ddos attacks at the application layer: Challenges and research perspectives for safeguarding web applications," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 661–685, 2018.

[56] H. Ballani, Y. Chawathe, S. Ratnasamy, T. Roscoe, and S. Shenker, "Off by default!" in *Proceedings of the Fourth Workshop on Hot Topics in Networking (HotNets-II)*. ACM, November 2005.

[57] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregates in the network," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 3, pp. 62–73, 2002.

[58] A. Mortensen, T. Reddy.K, F. Andreasen, N. Teague, and R. Compton, "DDoS Open Threat Signaling (DOTS) Architecture," RFC 8811, Aug. 2020.

[59] Z. Zhang, R. C. Aygun, G. Xiao, S. Song, E. Osterweil, A. Stavrou, and L. Zhang. (2023) Corpus of reviewed publications, and synthesized evaluation table. [Online]. Available: https://zhiyi-zhang.com/sok-paper-analysis/

[60] P. Ferguson and D. Senie, "network ingress filtering: defeating denial of service attacks which employ ip source address spoofing," *IETF, RFC 2827*, 2000.

[61] T. Killalea, "Recommended internet service provider security services and procedures," November 2000.

[62] K. Park and H. Lee, "On the effectiveness of route-based packet filtering for distributed dos attack prevention in power-law internets," *ACM SIGCOMM computer communication review*, vol. 31, no. 4, pp. 15–26, 2001.

[63] Jun Li, J. Mirkovic, Mengqiu Wang, P. Reiher, and Lixia Zhang, "Save: source address validity enforcement protocol," in *Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, 2002, pp. 1557–1566.

[64] S. Bosworth and M. E. Kabay, *Computer security handbook*. John Wiley & Sons, 2002.

[65] F. Baker and P. Savola, "Ingress filtering for multihomed networks," *IETF, RFC 3704*, 2004.

[66] X. Liu, A. Li, X. Yang, and D. Wetherall, "Passport: secure and adoptable source authentication," in *USENIX Symposium on Networked Systems Design and Implementation*, 2008, pp. 365–378.

[67] P. Du and A. Nakao, "Ddos defense deployment with network egress and ingress filtering," in *International Conference on Communications*. IEEE, 2010, pp. 1–6.

[68] G. Yao, J. Bi, and P. Xiao, "Source address validation solution with openflow/nox architecture," in *international conference on network protocols*. IEEE, 2011, pp. 7–12.

[69] Y. Afek, A. Bremler-Barr, and L. Shafir, "Network anti-spoofing with sdn data plane," in *INFOCOM*. IEEE, 2017, pp. 1–9.

[70] MANRS community. (2022) Mutually agreed norms for routing security (manrs). [Online]. Available: https://www.manrs.org/

[71] T. Dai and H. Shulman, "Smap: Internet-wide scanning for spoofing," in *Annual Computer Security Applications Conference*, 2021, pp. 1039–1050.

[72] X. Feng, Q. Li, K. Sun, Z. Qian, G. Zhao, X. Kuang, C. Fu, and K. Xu, "{Off-Path} network traffic manipulation via revitalized {ICMP} redirect attacks," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2619–2636.

[73] K. Man, Z. Qian, Z. Wang, X. Zheng, Y. Huang, and H. Duan, "Dns cache poisoning attack reloaded: Revolutions with side channels," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 1337–1350.

[74] A. Back *et al.*, "Hashcash-a denial of service counter-measure," 2002.

[75] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, "Captcha: Using hard ai problems for security," in *International conference on the theory and applications of cryptographic techniques*. Springer, 2003, pp. 294–311.

[76] W.-c. Feng, E. Kaiser, and A. Luu, "Design and implementation of network puzzles," in *Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 4. IEEE, 2005, pp. 2372–2382.

[77] S. H. Khor and A. Nakao, "Daas: Ddos mitigation-as-a-service," *IEEE/IPSJ International Symposium on Applications and the Internet*, pp. 160–171, 2011.

[78] B. Waters, A. Juels, J. Halderman, and E. Felten, "New client puzzle outsourcing techniques for dos resistance," *ACM Conference on Computer and Communications Security*, pp. 246–256, 01 2004.

[79] T. Anderson, T. Roscoe, and D. Wetherall, "Preventing internet denial-of-service with capabilities," *Computer Communication Review*, vol. 34, pp. 39–44, 01 2004.

[80] A. Yaar, A. Perrig, and D. Song, "Siff: a stateless internet flow filter to mitigate ddos flooding attacks," in *IEEE Symposium on Security and Privacy*, 2004, pp. 130–143.

[81] X. Yang, D. Wetherall, and T. Anderson, "A dos-limiting network architecture," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4, pp. 241–252, 2005.

[82] B. Parno, D. Wendlandt, E. Shi, A. Perrig, B. Maggs, and Y.-C. Hu, "Portcullis: protecting connection setup from denial-of-capability attacks," *SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 289–300, 2007.

[83] X. Yang, D. Wetherall, and T. Anderson, "Tva: A dos-limiting network architecture," *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, pp. 1267–1280, 2008.

[84] R. Stone *et al.*, "Centertrack: An ip overlay network for tracking dos floods." in *USENIX Security Symposium*, vol. 21, 2000, p. 114.

[85] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for ip traceback," in *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 2000, pp. 295–306.

[86] ——, "Network support for ip traceback," *IEEE/ACM transactions on networking*, vol. 9, no. 3, pp. 226–237, 2001.

[87] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer, "Hash-based ip traceback," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 3–14, 2001.

[88] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer, "Single-packet ip traceback," *IEEE/ACM Transactions on networking*, vol. 10, no. 6, pp. 721–734, 2002.

[89] A. Yaar, A. Perrig, and D. Song, "Pi: A path identification mechanism to defend against ddos attacks," in *Symposium on Security and Privacy*. IEEE, 2003, pp. 93–107.

[90] M. Handley and A. Greenhalgh, "Steps towards a dos-resistant internet architecture," in *Proceedings of the ACM SIGCOMM workshop on Future directions in network architecture*, 08 2004, pp. 49–56.

[91] X. Liu, X. Yang, and Y. Xia, "Netfence: preventing internet denial of service from inside out," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 4, pp. 255–266, 2010.

[92] M. H. Bhuyan, D. Bhattacharyya, and J. K. Kalita, "An empirical evaluation of information metrics for low-rate and high-rate ddos attack detection," *Pattern Recognition Letters*, vol. 51, pp. 1–7, 2015.

[93] J. Yu, H. Lee, M.-S. Kim, and D. Park, "Traffic flooding attack detection with snmp mib using svm," *Computer Communications*, vol. 31, no. 17, pp. 4212–4219, 2008.

[94] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[95] X. Yuan, C. Li, and X. Li, "Deepdefense: identifying ddos attack via deep learning," in *International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2017, pp. 1–8.

[96] J. Mirkovic, G. Prier, and P. Reiher, "Attacking ddos at the source," in *International Conference on Network Protocols*. IEEE, 2002, pp. 312–321.

[97] T. M. Gil and M. Poletto, "Multops: A data-structure for bandwidth attack detection." in *USENIX Security Symposium*, 2001, pp. 23–38.

[98] S. Yu, W. Zhou, R. Doss, and W. Jia, "Traceback of ddos attacks using entropy variations," *IEEE transactions on parallel and distributed systems*, vol. 22, no. 3, pp. 412–425, 2010.

[99] Y. Chen, K. Hwang, and W.-S. Ku, "Collaborative detection of ddos attacks over multiple network domains," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 12, pp. 1649–1662, 2007.

[100] J. François, I. Aib, and R. Boutaba, "Firecol: a collaborative protection network for the detection of flooding ddos attacks," *IEEE/ACM Transactions on networking*, vol. 20, no. 6, pp. 1828–1841, 2012.

[101] D. Turk, "Configuring bgp to block denial-of-service attacks," *IETF, RFC 3882*, September 2004.

[102] W. Kumari and D. McPherson, "Remote triggered black hole filtering with unicast reverse path forwarding (urpf)," *IETF, RFC 5635*, 2009.

[103] M. Nawrocki, J. Blendin, C. Dietzel, T. C. Schmidt, and M. Wählisch, "Down the black hole: Dismantling operational practices of bgp blackholing at ixps," in *Internet Measurement Conference*, 2019, pp. 435–448.

[104] F. Baker and P. Savola, "Ingress Filtering for Multihomed Networks," *IETF, RFC 3704*, Mar. 2004.

[105] N. Stamatelatos, "A measurement study of bgp blackhole routing performance," 2006. [Online]. Available: https://calhoun.nps.edu/handle/10945/2648

[106] J. Xiong, M. Wei, Z. Lu, and Y. Liu, "Warmonger: Inflicting denial-of-service via serverless functions in the cloud," in *ACM CCS*, 2021, pp. 955–969.

[107] C. Kustarz, L. B. Huston III, J. A. Simpson, J. E. Winquist, O. P. Barnes, and E. Jackson, "System and method for denial of service attack mitigation using cloud services," Aug. 30 2016, uS Patent 9,432,385.

[108] D. J. Smith, M. Glenn, J. A. Schiel, and C. L. Garner, "Network traffic data scrubbing with services offered via anycasted addresses," May 24 2016, uS Patent 9,350,706.

[109] P. Zilberman, R. Puzis, and Y. Elovici, "On network footprint of traffic inspection and filtering at global scrubbing centers," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 5, pp. 521–534, 2017.

[110] G. C. M. Moura, C. Hesselman, G. Schaapman, N. Boerman, and O. de Weerdt, "Into the ddos maelstrom: a longitudinal study of a scrubbing service," in *IEEE European Symposium on Security and Privacy Workshops*, 2020, pp. 550–558.

[111] D. Shapira, E. Cohen, T. Bronshtein, E. Leshem, and A. Ludmer, "Infrastructure distributed denial of service protection," July 23 2020, uS Patent App. 16/839,666.

[112] Z. Liu, H. Jin, Y.-C. Hu, and M. Bailey, "Middlepolice: Toward enforcing destination-defined policies in the middle of the internet," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 1268–1279. [Online]. Available: https://doi.org/10.1145/2976749.2978306

[113] K. J. Argyraki and D. R. Cheriton, "Active internet traffic filtering: Real-time response to denial-of-service attacks." in *USENIX annual technical conference, general track*, vol. 38, 2005.

[114] R. Chen, J.-M. Park, and R. Marchany, "Track: A novel approach for defending against distributed denial-of-service attacks," *Technical P˜ eport TR ECE—O6-02. Dept. of Electrical and Computer Engineering, Virginia Tech*, 2006.

[115] X. Liu, X. Yang, and Y. Lu, "To filter or to authorize: Network-layer dos defense against multimillion-node botnets," in *ACM SIGCOMM 2008 conference on Data communication*, 2008, pp. 195–206.

[116] A. D. Keromytis, V. Misra, and D. Rubenstein, "Sos: Secure overlay services," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 61–72, 2002.

[117] D. G. Andersen *et al.*, "Mayday: Distributed filtering for internet services." in *USENIX Symposium on Internet Technologies and Systems*, vol. 4, 2003.

[118] S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang, and X. S. Wang, *Moving target defense: creating asymmetric uncertainty for cyber threats*. Springer Science & Business Media, 2011, vol. 54.

[119] S. Jajodia, A. K. Ghosh, V. Subrahmanian, V. Swarup, C. Wang, and X. S. Wang, *Moving Target Defense II: Application of Game Theory and Adversarial Modeling*. Springer, 2012, vol. 100.

[120] Q. Jia, K. Sun, and A. Stavrou, "Motag: Moving target defense against internet denial of service attacks," in *22nd International Conference on Computer Communication and Network*. IEEE, 2013, pp. 1–9.

[121] Q. Jia, H. Wang, D. Fleck, F. Li, A. Stavrou, and W. Powell, "Catch me if you can: A cloud-enabled ddos defense," in *Annual IEEE/IFIP International Conference on Dependable Systems and Networks*. IEEE, 2014, pp. 264–275.

[122] H. Wang, Q. Jia, D. Fleck, W. Powell, F. Li, and A. Stavrou, "A moving target ddos defense mechanism," *Computer Communications*, vol. 46, pp. 10–21, 2014.

[123] P. Wood, C. Gutierrez, and S. Bagchi, "Denial of service elusion (dose): Keeping clients connected for less," in *Symposium on Reliable Distributed Systems (SRDS)*. IEEE, 2015, pp. 94–103.

[124] S. Venkatesan, M. Albanese, K. Amin, S. Jajodia, and M. Wright, "A moving target defense approach to mitigate ddos attacks against proxy-based architectures," in *Conference on communications and network security*. IEEE, 2016, pp. 198–206.

[125] D. McPherson, D. Oran, D. Thaler, and E. Osterweil, "Architectural considerations of ip anycast," *IETF*, no. 7094, Jan. 2014.

[126] G. C. Moura, R. d. O. Schmidt, J. Heidemann, W. B. de Vries, M. Müller, L. Wei, and C. Hesselman, "Anycast vs. ddos: Evaluating the november 2015 root dns," in *Proceedings of the ACM Internet Measurement Conference (IMC 2016). Santa Monica, CA, USA. November*, 2016.

[127] O. Spatscheck, Z. Al-Qudah, S. Lee, M. Rabinovich, and J. Van Der Merwe, "Multi-autonomous system anycast content delivery network," Dec. 10 2013, uS Patent 8,607,014.

[128] E. S.-J. Swildens, Z. Liu, and R. D. Day, "Global traffic management system using ip anycast routing and dynamic load-balancing," Mar. 8 2011, uS Patent 7,904,541.

[129] A. Rizvi, L. Bertholdo, J. Ceron, and J. Heidemann, "Anycast agility: Network playbooks to fight ddos," in *Proceedings of the 31st USENIX Security Symposium, page to appear. USENIX*, 2022.

[130] A. Yaar, A. Perrig, and D. Song, "Stackpi: New packet marking and filtering mechanisms for ddos and ip spoofing defense," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 10, pp. 1853–1863, 2006.

[131] Z. Liu, H. Namkung, G. Nikolaidis, J. Lee, C. Kim, X. Jin, V. Braverman, M. Yu, and V. Sekar, "Jaqen: A high-performance switch-native approach for detecting and mitigating volumetric ddos attacks with programmable switches," in *USENIX Security Symposium*, 2021.

[132] C. Chung, P. Khatkar, T. Xing, J. Lee, and D. Huang, "Nice: Network intrusion detection and countermeasure selection in virtual network systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 10, no. 4, pp. 198–211, 2013.

[133] T. Xing, D. Huang, L. Xu, C.-J. Chung, and P. Khatkar, "Snortflow: A openflow-based intrusion prevention system in cloud environment," in *second GENI research and educational experiment workshop*. IEEE, 2013, pp. 89–92.

[134] S. Lim, J. Ha, H. Kim, Y. Kim, and S. Yang, "A sdn-oriented ddos blocking scheme for botnet-based attacks," in *International Conference on Ubiquitous and Future Networks*. IEEE, 2014, pp. 63–68.

[135] C. Dillon and M. Berkelaar, "Openflow (d) dos mitigation," Technical Report (Feb 2014). http://www.delaat.net/rp/2013-2014/p42/report ..., Tech. Rep., 2014.

[136] B. Wang, Y. Zheng, W. Lou, and Y. T. Hou, "Ddos attack protection in the era of cloud computing and software-defined networking," in *IEEE International Conference on Network Protocols*, 2014, pp. 624–629.

[137] K. Giotis, C. Argyropoulos, G. Androulidakis, D. Kalogeras, and V. Maglaris, "Combining openflow and sflow for an effective and scalable anomaly detection and mitigation mechanism on sdn environments," *Computer Networks*, vol. 62, pp. 122–136, 2014.

[138] A. F. M. Piedrahita, S. Rueda, D. M. F. Mattos, and O. C. M. B. Duarte, "Flowfence: a denial of service defense system for software defined networking," in *Global Information Infrastructure and Networking Symposium*, 2015, pp. 1–6.

[139] T. Chin, X. Mountrouidou, X. Li, and K. Xiong, "Selective packet inspection to detect dos flooding using software defined networking (sdn)," in *IEEE International Conference on Distributed Computing Systems Workshops*, 2015, pp. 95–99.

[140] M. F. Hyder and T. Fatima, "Towards crossfire distributed denial of service attack protection using intent-based moving target defense over software-defined networking," *IEEE Access*, vol. 9, pp. 112 792–112 804, 2021.

[141] O. O. Olakanmi and K. O. Odeyemi, "Throttle: An efficient approach to mitigate distributed denial of service attacks on software-defined networks," *Security and Privacy*, p. e158, 2021.

[142] S. K. Fayaz, Y. Tobioka, V. Sekar, and M. Bailey, "Bohatei: Flexible and elastic ddos defense," in {*USENIX*} *Security Symposium*, 2015, pp. 817–832.

[143] M. Zhang, G. Li, S. Wang, C. Liu, A. Chen, H. Hu, G. Gu, Q. Li, M. Xu, and J. Wu, "Poseidon: Mitigating volumetric ddos attacks with programmable switches," in *NDSS*, 2020.

[144] C. Jung, S. Kim, R. Jang, D. Mohaisen, and D. Nyang, "A scalable and dynamic acl system for in-network defense," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 1679–1693.

[145] R. Klöti, V. Kotronis, and P. Smith, "Openflow: A security analysis," in *International Conference on Network Protocols*. IEEE, 2013, pp. 1–6.

[146] K. Benton, L. J. Camp, and C. Small, "Openflow vulnerability assessment," in *ACM SIGCOMM workshop on Hot topics in software defined networking*, 2013, pp. 151–152.

[147] S. Shin and G. Gu, "Attacking software-defined networks: A first feasibility study," in *ACM SIGCOMM workshop on Hot topics in software defined networking*, 2013, pp. 165–166.

[148] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, *et al.*, "Onix: A distributed control platform for large-scale production networks." in *OSDI*, vol. 10, 2010, pp. 1–6.

[149] P. Porras, S. Shin, V. Yegneswaran, M. Fong, M. Tyson, and G. Gu, "A security enforcement kernel for openflow networks," in *ACM SIGCOMM workshop on Hot topics in software defined networks*, 2012, pp. 121–126.

[150] S. Shin, V. Yegneswaran, P. Porras, and G. Gu, "Avant-guard: Scalable and vigilant switch flow management in software-defined networks," in *ACM SIGSAC CCS*, 2013, pp. 413–424.

[151] S. M. Mousavi, "Early detection of ddos attacks in software defined networks controller," Ph.D. dissertation, Carleton University, 2014.

[152] M. Liyanage, M. Ylianttila, and A. Gurtov, "Securing the control channel of software-defined mobile networks," in *International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE, 2014, pp. 1–6.

[153] H. Li, P. Li, S. Guo, and A. Nayak, "Byzantine-resilient secure software-defined networks with multiple controllers in cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 436–447, 2014.

[154] P. Dong, X. Du, H. Zhang, and T. Xu, "A detection method for a novel ddos attack against sdn controllers by vast new low-traffic flows," in *IEEE International Conference on Communications*, 2016, pp. 1–6.

[155] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "Ddos in the iot: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[156] J. Ioannidis and S. M. Bellovin, "Implementing pushback: Router-based defense against ddos attacks," *Network and Distributed System Security Symposium: NDSS '02*, 2002.

[157] G. Oikonomou, J. Mirkovic, P. Reiher, and M. Robinson, "A framework for a collaborative ddos defense," in *Annual Computer Security Applications Conference (ACSAC)*, 2006, pp. 33–42.

[158] D. Wagner, D. Kopp, M. Wichtlhuber, C. Dietzel, O. Hohlfeld, G. Smaragdakis, and A. Feldmann, "United we stand: Collaborative detection and mitigation of amplification ddos attacks at scale," in *ACM CCS*, 2021, pp. 970–987.

[159] B. Rodrigues, T. Bocek, A. Lareida, D. Hausheer, S. Rafati, and B. Stiller, "A blockchain-based architecture for collaborative ddos mitigation with smart contracts," in *Security of Networks and Services in an All-Connected World*, D. Tuncer, R. Koch, R. Badonnel, and B. Stiller, Eds. Cham: Springer International Publishing, 2017, pp. 16–29.

[160] CloudFlare. (2019) Cloudflare advanced ddos attack protection. [Online]. Available: https://www.cloudflare.com/ddos/

[161] C. Labovitz, "Internet traffic 2009-2019," *Presentation at NANOG*, vol. 76, pp. 9–12, 2019.

[162] D. Barrera, L. Chuat, A. Perrig, R. M. Reischuk, and P. Szalachowski, "The scion internet architecture," *Commun. ACM*, vol. 60, no. 6, p. 56–65, may 2017. [Online]. Available: https://doi.org/10.1145/3085591

[163] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, 2009, pp. 1–12.

[164] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, *et al.*, "Named data networking," in *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3. ACM, 2014, pp. 66–73.

[165] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, and L. Zhang, "A case for stateful forwarding plane," *Computer Communications*, vol. 36, no. 7, pp. 779–791, 2013.

[166] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, and L. Zhang, "Interest flooding attack and countermeasures in named data networking," in *IFIP Networking Conference, 2013*. IEEE, 2013, pp. 1–9.

[167] A. Compagno, M. Conti, P. Gasti, and G. Tsudik, "Poseidon: Mitigating interest flooding ddos attacks in named data networking," in *Conference on Local Computer Networks (LCN)*. IEEE, 2013, pp. 630–638.

[168] H. Dai, Y. Wang, J. Fan, and B. Liu, "Mitigate ddos attacks in ndn by interest traceback," in *IEEE International Conference on Computer Communications (INFOCOM WKSHPS)*. IEEE, 2013, pp. 381–386.

[169] H. Salah and T. Strufe, "Evaluating and mitigating a collusive version of the interest flooding attack in ndn," in *IEEE Symposium on Computers and Communication (ISCC)*. IEEE, 2016, pp. 938–945.

[170] Y. Xin, Y. Li, W. Wang, W. Li, and X. Chen, "A novel interest flooding attacks detection and countermeasure scheme in ndn," in *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–7.

[171] Y. Nakatsuka, J. L. Wijekoon, and H. Nishi, "Frog: A packet hop count based ddos countermeasure in ndn," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2018, pp. 00 492–00 497.

[172] A. Benmoussa, A. el Karim Tahari, C. A. Kerrache, N. Lagraa, A. Lakas, R. Hussain, and F. Ahmad, "Msidn: Mitigation of sophisticated interest flooding-based ddos attacks in named data networking," *Future Generation Computer Systems*, vol. 107, pp. 293–306, 2020.

[173] Z. Zhang, V. Vasavada, S. K. R. Kakarla, A. Stavrou, E. Osterweil, and L. Zhang, "Expect more from the networking: Ddos mitigation by fitt in named data networking," *arXiv preprint. 1902.09033.*, 2021.

[174] R. A. Al-Share, A. S. Shatnawi, and B. Al-Duwairi, "Detecting and mitigating collusive interest flooding attacks in named data networking," *IEEE Access*, vol. 10, pp. 65 996–66 017, 2022.

[175] Z. Xu, X. Wang, and Y. Zhang, "Towards persistent detection of ddos attacks in ndn: A sketch-based approach," *IEEE Transactions on Dependable and Secure Computing*, 2022.

[176] K. Nichols, "Defined-trust limited domains," *Technical Disclosure Commons*, 2023.

[177] B. Carpenter and B. Liu, "Limited domains and internet protocols," *IETF, RFC 8799*, 2020.