Zero-setup Intermediate-rate Communication Guarantees in a Global Internet

Marc Wyss
ETH Zurich

Adrian Perrig
ETH Zurich

Abstract

Network-targeting volumetric DDoS attacks remain a major threat to Internet communication. Unfortunately, existing solutions fall short of providing forwarding guarantees to the important class of *short-lived intermediate-rate communication* such as web traffic in a secure, scalable, light-weight, low-cost, and incrementally deployable fashion. To overcome those limitations we design Z-Lane, a system achieving those objectives by ensuring bandwidth isolation among authenticated traffic from (groups of) autonomous systems, thus safeguarding intermediate-rate communication against even the largest volumetric DDoS attacks. Our evaluation on a global testbed and our high-speed implementation on commodity hardware demonstrate Z-Lane's effectiveness and scalability.

1 Introduction

Network-targeting distributed denial-of-service (DDoS) attacks flood routers and links with massive amounts of traffic, resulting in congestion and the subsequent dropping of legitimate packets along their communication path. Despite decades of research, volumetric DDoS attacks against network infrastructure remain a threat to the availability of Internet communication; both the number of attacks and their volume have been increasing over the past years with a large fraction of attacks relying on address spoofing [15, 52]. With botnets, in particular Mirai variants, and pulse-wave attacks on the rise, DDoS threats are becoming more pronounced [16]. Home connections with 10 Gbps speeds and botnets composed of powerful cloud VMs are simplifying the execution of volumetric DDoS attacks and render them increasingly potent [17].

In this work, we specifically focus on ensuring communication reliability for *short-lived intermediate-rate traffic*, including DNS communication, legitimate command and control systems, and access to websites. Those applications typically require rates ranging from individual packets to a few hundreds of kilobits per second, occasionally reaching up to several megabits per second at most, and often persist for the duration of a few seconds only. While short-lived traffic constitutes the majority of concurrent flows, it accounts for only little total traffic volume, i.e., less than 10% in terms of both packets and bytes per second [69]. Still, such communication is delay-critical, and therefore protection against volumetric DDoS should be *available immediately* and thus readily protect a flow's first packet(s), instead of being reactive.

As we elaborate in Section 8, existing solutions, both from industry and academia, have proven unsatisfactory in addressing this problem. Industry systems often grapple with scalability issues, substantial management overhead, and high cost, while research-based systems frequently resort to heuristics, work only reactively, cannot protect the whole forwarding path, cause latency inflation, or remain vulnerable to spoofed packet source addresses. Even next-generation bandwidth reservation systems designed to explicitly allocate bandwidth on specific inter-domain paths fall short of effectively addressing this challenge: the time required to establish a new reservation introduces substantial delays, which significantly outweigh the short-lived duration of the actual traffic transmission. Furthermore, the minimum validity period of a reservation typically spans over tens of seconds, resulting in wastage of reserved bandwidth for short-lived application traffic.

Apparently, protecting short-lived intermediate-rate communication in the global Internet is challenging and calls for novel, low-cost defense mechanisms that work proactively instead of reactively, provide communication guarantees rooted in solid theoretical foundations as opposed to heuristics, scale to large topologies, are fully resilient to source address spoofing, and do not introduce additional latency.

Our key insight lies in recognizing that (i) our target traffic class necessitates only a minimal amount of traffic, rendering a small pre-configured bandwidth allocation at routers sufficient to guarantee its forwarding, and (ii) effective enforcement of bandwidth isolation by a congested router requires the ability to ascertain the absence of spoofed source addresses.

Recent advancements in path-aware architectures have introduced mechanisms for authenticating traffic sources to every on-path autonomous system (AS)[1] [42], with in-network filtering of spoofed traffic and thwarting reflection and amplification attacks as their primary objectives. However, we observe that these mechanisms possess untapped potential, as they can enable *meaningful* bandwidth allocations at routers, aligning with the requirements of our insight. Given that the security of these mechanisms is rooted in AS-level shared secret keys, our proposal centers on protecting short-lived intermediate-rate communication through *AS-level bandwidth isolation applied to source-authenticated traffic*, thus avoiding the need for the setup of measures such as bandwidth reservations prior to actual traffic transmission ("zero-setup").

Still, numerous challenges persist. These range from efficiently monitoring and scheduling traffic originating from tens of thousands of ASes without requiring excessive queues or memory at routers, to protecting against malicious end hosts that may drain their ASes' guaranteed rates. To address these scalability and security challenges, we adopt a hierarchical approach, wherein we organize ASes into groups that share a uniform trust environment. Consequently, traffic originating from within a router's designated group can be granted per-AS forwarding guarantees, while traffic from external sources can be scheduled at the AS group level. Furthermore, an AS can extend its guaranteed rates to its end hosts, thus ensuring DDoS protection for individual users.

The compilation of these ideas results in the design of Z-Lane, a system providing zero-setup communication guarantees for intermediate-rate Internet traffic. Z-Lane consists of (i) a gateway to be deployed at source ASes and (ii) an efficient scheduling mechanism to implement AS-level bandwidth isolation at border routers. Due to ubiquitous source authentication, Z-Lane can offer fundamental communication guarantees, avoiding the reliance on heuristic methods. If globally deployed, Z-Lane can defend the many end hosts in the Internet against even the largest DDoS attacks without introducing significant latency, thus extending its applicability beyond large-scale enterprises. Our main contributions are:

- The design of Z-Lane, a system providing zero-setup communication guarantees for intermediate-rate traffic.
- A formal analysis showing that Z-Lane can protect such traffic against volumetric DDoS attacks, emphasized through a comparative evaluation with related work.
- The deployment of Z-Lane on a global testbed to show its practical viability and incremental deployability.
- The implementation of a high-performance version of Z-Lane in DPDK that generates, validates, and schedules traffic at 160 Gbps line rate on commodity hardware.

2 Background

We build Z-Lane, our proposed system described in Section 4, on top of SCION [14], a path-aware Internet architecture, and use EPIC [42] to implement per-packet source authentication.

SCION. As in BGP, the building blocks of the SCION network architecture are autonomous systems (ASes). However, as a fundamental concept distinguishing SCION from BGP, ASes are grouped into *isolation domains (ISDs)*. An ISD is a set of ASes that span a uniform trust environment or a common jurisdiction. In practice, an ISD might comprise ASes of providers associated with the same country or organisations such as banks or health care services. A subset of ASes in an ISD are *core ASes*, which link to core ASes of other ISDs to enable global connectivity; all other ASes in an ISD are referred to as *non-core ASes*. An AS network typically consists of multiple routers, where some of them are internal routers and some of them are border routers. A SCION border router has one internal interface for connectivity (via internal routers) to AS-local infrastructure services, end hosts, and to other border routers of the same AS, and one or multiple external interfaces, which provide connectivity to other ASes. Switching from BGP to SCION requires replacing existing border routers; all internal routers can be reused.

Research on SCION is facilitated by its open-source implementation [60] and SCIONLab [2, 41], a global SCION research testbed; several proposed systems already made it from research into SCION protocol extensions. SCION is being deployed in the real world [4, 6, 32, 40, 65–67] and has reached a Technology Readiness Level (TRL) of 9 [12]. There is work in progress to standardize SCION at IETF [19].

Z-Lane relies on SCION due to its (i) availability properties and (ii) AS groupings. BGP has inherent limitations that can compromise availability, from expensive reconvergence processes to path hijacking vulnerabilities enabling traffic redirection and packet dropping. In contrast, SCION as a multipath architecture offers a more robust solution by allowing traffic sources to immediately switch to other paths in response to on-path router or link failures (without requiring reconvergece) and is resilient to hijacking attacks by design. Operating independently of BGP, SCION ensures that vulnerabilities or attacks in BGP do not affect its own security. Furthermore, to achieve scalability to tens of thousands of ASes, Z-Lane needs a way to group ASes based on mutual trust; such a grouping is already implemented through ISDs in SCION. The grouping must not be arbitrary but trust-based, as outside the group, Z-Lane-enabled routers only enforce bandwidth isolation for the whole group rather than for individual ASes, meaning that ASes inside the group share their fate regarding their guaranteed forwarding rates. As SCION packets always contain the source AS and ISD, Z-Lane routers can avoid keeping a map from IP prefixes to ASes and ISDs.

[1] An AS is a network under a common administrative control such as an Internet service provider (ISP), a company, or a university.

Table 1: Overview of Z-Lane and its key technologies.

Protocol	Description
SCION	Internet architecture that groups ASes into ISDs. It ensures rapid failover via multipathing and resilience against path hijacking through path authentication and authorization.
DRKey	Establishes symmetric keys between ASes and endpoints. Keys can be dynamically derived at routers.
EPIC	Prevents spoofing and reflection attacks by offering source authentication for both routers and destinations.
Z-Lane	Protects short-lived, intermediate-rate traffic from DDoS via configurable per-AS/per-ISD bandwidth isolation.

EPIC. EPIC ensures that every packet's length and origin, i.e., the source host's ISD, AS, and address, can be efficiently verified by all on-path border routers and the destination host. This enables EPIC to effectively counter attacks that exploit spoofed addresses for reflection and amplification early on the communication path. In EPIC, a source host proves its authenticity to on-path border routers by generating a per-packet hop validation field (HVF) for each AS along the selected path. These HVFs are then included in the SCION packet header, replacing the existing hop fields. A HVF serves as a message authentication code (MAC) and is computed using a host-level symmetric key shared between the source host and a specific on-path AS. These keys are established through the DRKey system [38, 57], enabling routers to efficiently derive the keys within tens of nanoseconds from a master key, avoiding to store per-host key tables. While DRKey creates AS-level keys as well, these are only accessible to AS infrastructure services and not to end hosts. Keys established with DRKey are valid for one day and can be renewed before their expiration; they are requested before any actual communication takes place and can be reused for all EPIC-authenticated flows. Through the deployment of a duplicate-suppression system, EPIC allows transit ASes to also filter replayed packets. Details on DRKey and EPIC can be found in Appendices A and B.

We rely on EPIC's efficient source authentication mechanism proven secure in a strong attacker model to prevent packet spoofing in Z-Lane. We use the EPIC system in particular due to (i) lack of alternatives and (ii) its integration into SCION. As we elaborate in Section 8, to the best of our knowledge EPIC is the only system that authenticates every packet's source (and its length) to every on-path AS, while operating at line rate and being stateless at routers, and thus scales at border routers to arbitrarily large topologies. EPIC's verification at all on-path routers contrasts other source authentication schemes where routers only probabilistically verify a subset of all received packets. Also, EPIC is specifically designed for path-aware networks like SCION. To authenticate its traffic, a source host in EPIC requires explicit knowledge regarding all on-path ASes. This requirement underscores the significance of SCION's path transparency, which is lacking in BGP.

Table 1 summarizes Z-Lane and its key technologies.

3 Requirements and Assumptions

In this section, we outline the problem we aim to solve, the requirements that any solution attempting to do so needs to satisfy, as well as our threat- and network models.

3.1 Problem Statement

Our objective is to provide communication guarantees to short-lived, intermediate-rate inter-domain traffic of protocols such as DNS, HTTP(S), or SSH despite volumetric DDoS attacks. Those protocols are characterized by their relatively low traffic volumes typically ranging from a single packet to a few megabits per second, and their short duration, from the time needed for a single round-trip to a few seconds. We focus on the availability of this type of communication across the Internet, i.e., on mitigating the impact of volumetric DDoS attacks on the forwarding behavior of legitimate traffic at routers. While EPIC mitigates attacks leveraging spoofed addresses for reflection and amplification, vulnerabilities persist in the face of attacks originating *from individual sources generating immense traffic volumes or from botnets where multiple distinct end hosts transmit traffic at lower rates* [36, 48, 64]. Such attacks can disrupt connectivity among target links and thus even disconnect whole regions and are often regarded as some of the most challenging threats to availability [70]. DDoS attacks targeted at end hosts or exploiting weaknesses in higher-layer protocols are not in the scope of this work.

3.2 Requirements

We require our system to:

R1 Guarantee forwarding of intermediate-rate inter-domain traffic despite arbitrarily large volumetric DDoS attacks.

R2 Provide guarantees also in Internet topologies comprising hundreds of thousands of ASes and billions of end hosts.

R3 Not require any setup process before end hosts can send actual traffic ("zero-setup"). This distinguishes our setting from inter-domain bandwidth reservation systems, which have an inherent reservation setup phase (Section 8).

R4 Avoid latency inflation, such as from added packet processing, queuing, or rerouting delays.

R5 Not incur substantial financial costs, including operational costs for providers and end domains.

Besides those main requirements, an ideal solution should attain high rates for traffic generation and forwarding while minimizing hardware demands, be incrementally deployable at border routers, and maintain a simple design relying on established primitives for effortless deployment and configuration. To the best of our knowledge, no currently existing system is able to satisfy all of those requirements (Section 8).

3.3 Assumptions

Adversary Model. For communication to be possible between some source and some destination host, there must exist at least one adversary-free path over the Internet (i.e., all on-path entities are benign), as attackers could trivially drop packets to prevent any communication. SCION usually provides dozens of paths to end hosts, allowing avoidance of paths where communication is not possible or has low QoS. In our model, on-path attackers can nevertheless observe, inject, and replay packets on inter-AS links.

Even when communication occurs on a path with benign entities only, off-path entities may attempt to hinder it, such as by creating congestion on forwarding devices along the route. Our adversary model allows off-path attackers to exhibit arbitrary behavior. In particular, off-path attackers can also control ASes, including compromised services, end hosts, and routers, where the attacker can also have access to AS key material. Those assumptions are in line with the attacker model of other work related to DDoS defense [31, 75].

Network Characteristics. We assume that adjacent ASes connect through direct links. If they instead connect through lower layer interconnects at an IXP, e.g., in a colocation facility, we expect the IXP to implement adequate isolation measures to protect each inter-AS peering against congestion in its network. Alternatively, an IXP may itself implement Z-Lane and explicitly participate in routing by registering as an AS. We further assume that every network link is connected to routers that are capable of handling traffic rates matching the link's capacity so that packets are never dropped at ingress.

4 Z-Lane System Description

This section describes Z-Lane, our proposed DDoS-defense system satisfying the requirements in Section 3.2.

4.1 Overview

With per-AS bandwidth isolation implemented through DDoS-resistant and adjustable *forwarding rate lower bounds for (groups of) ASes*, Z-Lane provides a novel perspective on how to achieve communication guarantees in a global Internet.

Guarantees for (Groups of) ASes. Z-Lane introduces the concept of bandwidth isolation at the level of (trust-based groups of) AS. At a border router, traffic originating from within its group gets per-AS forwarding guarantees, and traffic from outside is scheduled at the level of AS groups.

This design decision is motivated by (i) *scalability*, as AS-level traffic scheduling is challenging, whereas for groups of ASes it becomes feasible (analyzed in Section 4.5), (ii) *hierarchy*, as ASes can further pass on the AS-level lower bounded rates to their end hosts [75], (iii) *security*, because Sybil attacks, where an attacker creates many instances of a source in an attempt to get a higher share of a resource, are much harder when those sources are ASes [18], and (iv) *traffic stability*, i.e., aggregated traffic from one or multiple ASes naturally exhibits greater stability over time compared to flow or host-level traffic [55].

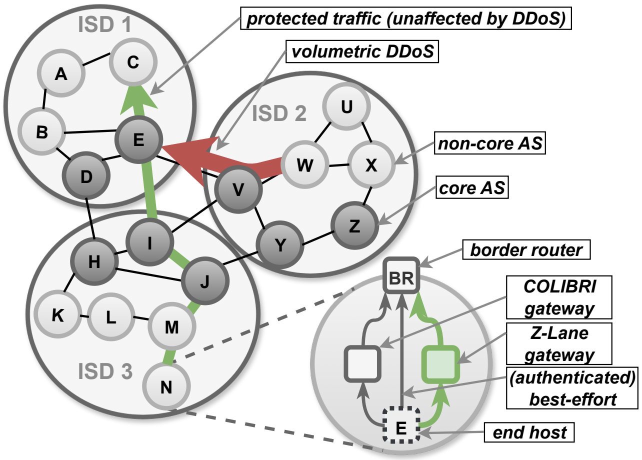

Figure 1: Overview. End hosts send short-lived traffic to the Z-Lane gateway, reservation traffic to the COLIBRI gateway, and all other traffic as authenticated best-effort directly to the border router. Z-Lane (and COLIBRI) traffic is protected at routers along the forwarding path from volumetric DDoS.

We instantiate the AS groupings with SCION isolation domains (ISDs). In the example in Figure 1, AS N of ISD 3 sends traffic to AS C of ISD 1, where its guaranteed rate at border routers protects this traffic against volumetric attacks from ISD-internal off-path attackers, e.g., AS L targeting the link between AS M and AS J, and ASes in off-path ISDs, e.g., AS W targeting the link between AS E and AS C.

Configurable Rates. Not all ASes generate identical traffic volumes, therefore Z-Lane allows a transit AS to configure different guaranteed rates for different ASes and ISDs. The configuration can for example be learned based on ASes' and ISDs' demands during normal operating conditions, such that the corresponding guaranteed rates can later automatically be enforced during periods of active attacks, enabling an AS to send traffic at the same rate as on previous days (Section 4.6).

If the guaranteed rate configured for a source AS is higher than its demand, its intermediate-rate traffic is completely unaffected by volumetric DDoS attacks. Z-Lane ensures that ASes and ISDs can send *at least* at their guaranteed rates, as it also forwards traffic exceeding their rate as best-effort. As long as there is remaining capacity, ASes and ISDs can thus overuse their guaranteed rates, so that no bandwidth is ever wasted. This implies that even if the allocated rate for a source AS is less than its demand, the AS can still communicate at its desired rate when there is no congestion. As we demonstrate in Section 5, the AS can also send data at the guaranteed rate during volumetric DDoS attacks, where communication would otherwise be impossible without the deployment of

Z-Lane. The decision how to configure the rates is ultimately up to the network operator and, importantly, does not require any inter-domain coordination. Due to the aggregation of ASes into ISDs, configurations remain manageable even if the Internet grows to hundreds of thousands of ASes.

End Host Guarantees. Z-Lane lets end hosts, more specifically their applications, define what traffic should be sent with forwarding guarantees, and what traffic should be forwarded over best-effort. Still, to protect against malicious end hosts, their AS has the ultimate authority in this matter and can reclassify traffic to be sent as best-effort only. This protection is implemented through a Z-Lane gateway, which schedules end host traffic and authenticates it towards on-path routers using a secret key not known to the end hosts. How traffic is scheduled is up to the AS operator; configurations can range from fair sharing to prioritizing certain traffic from critical AS services like routing or time synchronization. We emphasize that, to avoid any setup overhead (R3), neither ISDs, nor ASes or end hosts explicitly learn their configured rate; instead, end hosts implicitly discover their allowed rate through existing mechanisms like congestion control.

Compatibility with Other Systems. Bandwidth reservation systems cannot provide zero-setup communication guarantees and are therefore not suitable to protect short-lived intermediate-rate communication (Section 8). Still, we design Z-Lane to seamlessly coexist with them, as they complement our work by effectively protecting *non-setup-critical, high-volume communication* such as from video conferencing. We choose COLIBRI [27] as a reservation system instantiation, but other systems could be deployed as well. To prevent attacks targeting DRKey's AS-level key exchange, which is a fundamental requirement for EPIC, our design also ensures compatibility with the DoCile system [74], which leverages dedicated channels between neighboring ASes to successfully bootstrap the key exchange even under DDoS.

We therefore consider the following four types of inter-domain traffic: COLIBRI reservation traffic, DoCile's neighbor-based communication, authenticated traffic from EPIC, and unauthenticated SCION traffic.

4.2 Source Authentication

Z-Lane employs EPIC for authenticating traffic sources to border routers, allowing every border router to verify the authenticity of every received packet. An important insight in the design of Z-Lane is that efficient and reliable source authentication as provided by EPIC allows for *meaningful source-based traffic control at border routers*. The realization of this idea has not been possible so far because previous source authentication mechanisms would cause excessive communication or computation overhead and therefore impede deployment, or were based on heuristics or probabilities, and would thus fail to reliably distinguish between authentic and spoofed addresses (Appendix H). Z-Lane is the first system to explore the use of comprehensive source authentication to protect the availability of short-lived intermediate-rate Internet traffic – with *EPIC's security rooted in AS-level secret keys, it integrates seamlessly into Z-Lane*.

We want to highlight that EPIC together with a fairness mechanism provided by some congestion control algorithm, i.e., without any guaranteed rates, would not be enough in our threat model, as an attacker would just not respect the algorithm's feedback and instead keep sending traffic at high rates, or leverage a botnet to create many low-volume flows.

4.3 End Host Traffic Generation

End hosts, i.e., their applications, can choose among several mechanisms on how their traffic is forwarded (Figure 1). For long-term traffic they request a bandwidth reservation and use it by sending their COLIBRI traffic class packets through the COLIBRI gateway. While the overhead for requesting a reservation is significant, the result is a fixed amount of bandwidth that is exclusively reserved along the communication path. In a similar way, applications send short-lived intermediate-rate traffic using the EPIC traffic class over the Z-Lane gateway, where traffic is forwarded immediately without any delay (requirement R3), but without the applications knowing the concrete rates. In both cases traffic is protected against congestion on the communication path. The default option is for end hosts to send their traffic using the EPIC traffic class directly to a border router of their AS, where they are forwarded along the path using best-effort. This option is useful for non-latency-critical communication such as file downloads, or for long-term traffic for which no reservation is available, which can for example happen if the end host has already created a large number of reservations and gets denied from creating even more. Z-Lane envisages unauthenticated SCION traffic to be sent only in scenarios where it is not otherwise possible, e.g., if an AS needs to request shared keys using DRKey from another AS for the first time.

4.4 Z-Lane Gateway

ASes use the gateway to control the traffic volumes that their end hosts (incl. AS infrastructure services) are allowed to send using Z-Lane, which serves the primary purpose of protecting benign from malicious or compromised end hosts.

For end host traffic complying with the allowed rate, the gateway sets a QoS flag in the EPIC header, which indicates to on-path routers that the corresponding packets should be forwarded using the AS' guaranteed rate. If an end host's packet exceeds the allowed rate at the gateway, then either (i) the QoS flag is not set (or removed, if it was already set by the end host), meaning that those packets will be treated as best-effort, or (ii) the packets are dropped, depending on the AS' policy. In contrast to best-effort EPIC packets generated at end hosts, which are authenticated using host-level keys, the Z-Lane gateway further authenticates packets from end hosts using AS-level keys, which are not available to the end hosts, preventing end hosts from setting the QoS flag themselves. Details of DRKey, EPIC, and the authorization procedure at the gateway, are described in Appendices A to C.

Attack Prevention. Compared to a design where end hosts can set and authenticate the QoS flag themselves, the Z-Lane gateway, which further rate-limits the traffic and authenticates it with the AS-level key, provides better protection against misbehaving end hosts. In particular, it prevents malicious, compromised, or misconfigured end hosts from executing volumetric DDoS attacks aimed at overusing its AS' guaranteed forwarding rate at on-path border routers, thus disrupting communication of legitimate end hosts in its own AS.

If the gateway would not authenticate traffic using the AS-level key, malicious end hosts could leak their host-level keys to other entities in the Internet to enable them to launch DDoS attacks using packets with a spoofed source. However, with our design, such leakage does not provide any benefits to such entities, as they would still need to either (i) obtain access to the end host AS' keys, or (ii) route traffic through the end host AS' gateway, both of which are inaccessible to them.

Incentivizing Compliance. In principle, end hosts can send arbitrary application traffic to the gateway, as the gateway does not inspect packet payloads. However, the gateway disincentivizes end hosts from using Z-Lane also for high-volume application traffic, which should instead be sent either as authenticated best-effort or over COLIBRI reservations: Because the gateway only adds the QoS flags to packets up to a certain rate, and because it cannot distinguish between low-volume short-term and high-volume long-term traffic, the QoS flag will be randomly set on a small fraction of both types of traffic, thereby foiling the forwarding guarantees of both short-term and long-term traffic.

4.5 Border Router Traffic Scheduling

An essential part of Z-Lane is the enforcement of the configured AS- and ISD-level traffic rates at border routers. The challenge is to find a solution that is based on simple and well-known mechanisms (to facilitate adoption), requires minimal state and queues (to allow deployment on different devices), can forward traffic at high rates (to keep up with traffic rates on inter-domain links), and scales to the size of the Internet (requirement R2). In the following, we present our solution to overcome those challenges; an overview is given in Figure 2.

Queues. We introduce the following five different per-router-interface egress queues. The *reservation* queue (R) contains COLIBRI packets belonging to a bandwidth reservation. The *neighbor* queue (N) is used for DoCile packets originating from or destined to a neighboring AS.[2] The *guaranteed* queue (G) is meant for AS-level authenticated EPIC traffic where the QoS flag is set. Host-level authenticated EPIC traffic or AS-level EPIC traffic where the QoS flag is not set goes to the *authenticated* queue (A). Lastly, the *unauthenticated* queue (U) contains standard, i.e., unauthenticated, SCION packets. Packets from the queues R, N, and G are forwarded preferentially, while A and U are best-effort queues.

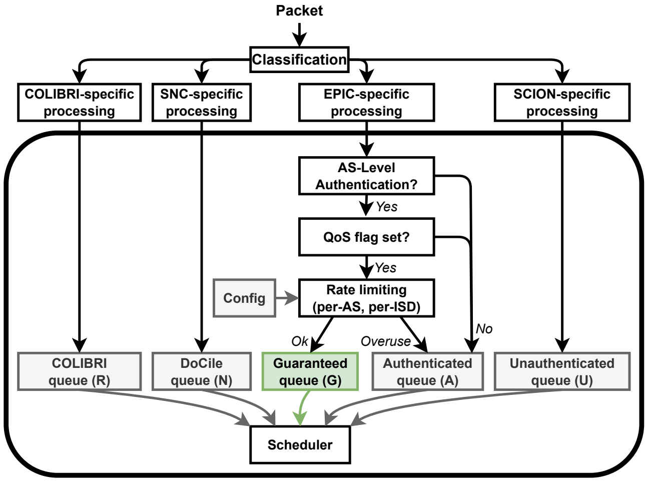

Figure 2: Z-Lane packet processing at the border router.

Packet Processing. COLIBRI, DoCile, host-level authenticated EPIC, and unauthenticated SCION packets are processed by the border router according to their protocol, and in case they are not dropped during this step, they are directly put into their corresponding queue (R, N, A, or U). AS-level authenticated EPIC packets are, after protocol-specific processing, further classified before they are added to a specific queue. In particular, packets with the QoS flag set that originate from an AS inside the border router's ISD are per-source-AS rate-limited according to a previously defined configuration. Similarly, packets with the QoS flag set, which originate from an AS outside the border router's ISD are rate-limited on a per-source-ISD basis. When a packet causes a violation of this rate, it is enqueued at A. Otherwise, the packet is added to G. This ensures that traffic up to the configured rate is guaranteed to be forwarded, and everything above this rate is forwarded as best-effort. *Contrary to intuition, rate-limiting EPIC traffic does therefore not serve the purpose of upper bounding the egress rates of ASes and ISDs, but instead to provide them with a lower bound.* If a packet cannot be added to a queue because the queue is already full, then the packet is dropped.[3] This can only happen for the best-effort queues A, and U however. The per-AS and per-ISD rate limits at the router (and also the end host rate limits at the gateway) can be efficiently implemented using token buckets [49]. Appendices E and F explain why we neither use probabilistic

[2] DoCile is needed to bootstrap the keys used in EPIC under DDoS where the guaranteed Z-Lane rate for the neighboring AS cannot be used yet.

[3] Instead of tail drop, other disciplines such as RED [24] could be employed.

Table 2: Scheduling bounds for the different queues with example instantiations as percentages of the total egress capacity. Their sum must not exceed 100 %.

Queue	Lower/Upper Bound	Example
Reservation (R)	μ_R	50 %
Neighbor (N)	μ_N	10 %
Guaranteed (G)	μ_G	20 %
Queue	Lower Bound	Example
Authenticated (A)	μ_A	15 %
Unauthenticated (U)	μ_U	5 %

monitoring nor neighbor-based scheduling, even though those mechanisms could further reduce Z-Lane's memory footprint.

The border router not only forwards traffic, but also creates new packets. This happens for (i) bidirectional forwarding detection (BFD) [1], which is used to detect faults in the links between routers, and (ii) SCMP [68], where the router responds to, e.g., traceroute requests. BFD packets are added to N, while SCMP packets are enqueued at U.

Scheduling. A router egress scheduler determines which packets to forward on its link, which is only relevant when there is congestion, i.e., when traffic exceeds the interface's capacity. Without congestion, no packets are dropped.

There are two requirements that the scheduler must satisfy. First, it must provide a lower bound on the forwarding rate to A and U. If packets are added to one of those queues at a rate α, where the lower bound provided to this queue is μ, then traffic from this queue must be forwarded at a rate greater or equal to $\min(\alpha, \mu)$. This requirement ensures that best-effort traffic will not starve. Second, the forwarding rates for R, N, and G must both be lower- and upper bounded.If traffic arrives at such a queue at a rate α, where the bound provided to this queue is μ, then traffic from that queue must be forwarded at a rate of exactly $\min(\alpha, \mu)$. Both requirements together imply that in case R, N, or G are not fully utilized, the remaining egress capacity can be used for traffic from A and U. The queues and their bounds are summarized in Table 2. An efficient scheduler that satisfies those two requirements by enforcing the given bounds is the hierarchical token bucket (HTB) scheduler [11]. If the deployed COLIBRI protocol implementation already guarantees that its upper bound μ_R is never exceeded (for DoCile and Z-Lane this is always the case), then also schedulers such as the deficit round robin (DRR) scheduler [9], the weighted fair queueing (WFQ) scheduler [20, 53], or the enhanced transmission selection (ETS) scheduler [10] can be used. Whether μ_R is already enforced by COLIBRI depends on the concrete protocol implementation, e.g., reservation monitoring in COLIBRI could be performed deterministically or probabilistically, where only the former implementation guarantees this upper bound.

Design Rationale. Optimizing the number of queues and the state required for rate limiting is important to enable implementations on high-speed hardware [34]. Our presented design uses five static queues (R, N, G, A, and U) plus a variable number of token buckets for per-AS and per-ISD rate limiting of guaranteed traffic at each interface, which constitutes a practically feasible, efficient, and simple-to-analyze solution to enforce Z-Lane's guarantees. However, other approaches satisfying Z-Lane's requirements are possible as well. Figure 3 compares (i) our proposal with designs based on (ii) a single queue G and per-AS token buckets, (iii) per-AS and per-ISD queues, and (iv) per-AS queues only, where the latter two do not require any token buckets. Monitoring 10^5 globally distributed ASes at a single egress interface for example requires (i) in Z-Lane only 5 static queues and 5.3 kB of memory, and otherwise (ii) 5 queues and 800 kB of memory, (iii) 638 queues, or (iv) 10^5 queues, respectively, which highlights the benefits of ISD-level traffic control. For (iii) and (iv), additional memory is needed for the queues and the scheduler, and the queues are also more difficult to add, modify, and remove at runtime. Still, approach (iii) has a manageable number of queues and could potentially be implemented with, e.g., SENIC [56], which keeps queues in memory and schedules—and thereby rate-limits—packets in the network interface card (NIC). The token bucket memory requirements are taken from Appendix D, where we describe an optimized rate limiter which reduces the memory overhead compared to current state-of-the-art implementations by 60-86%.

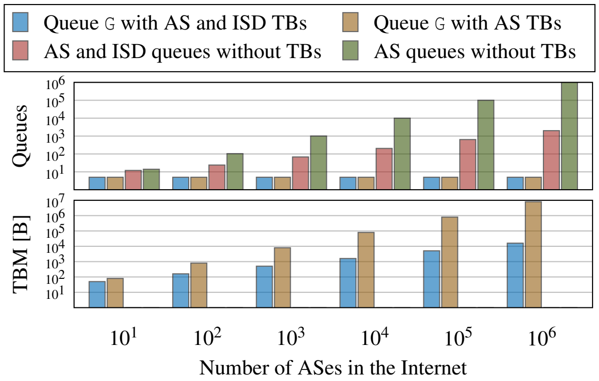

Figure 3: Nr. queues vs. token bucket memory (TBM) for different traffic control designs. Z-Lane implements a single queue G with per-AS and per-ISD token buckets (TBs).[4]

4.6 Configuration

The gateway configuration allows to set the rate up to which the QoS flag should be set of individual, or—based on IP prefixes—groups of end hosts and AS services. Similarly, the scheduling bounds μ for the different queues as well as the guaranteed forwarding rates for ASes and ISDs at border router interfaces can be configured by the router's network operator (Figure 2). The per-AS and per-ISD rates can either be absolute (e.g., 2 Gbps) or relative (e.g., 5 %) to the fraction μ_G of egress capacity dedicated to G. Thereby, the sum of the

[4] For simplicity, the analysis assumes that in a SCION Internet consisting of N ASes, an average ISD comprises approximately $\sqrt{N}$ ASes.

rates in relative terms must not exceed 100 %.

To simplify integration into an AS' network, Z-Lane has a default router configuration that ensures every AS and ISD receives an equal rate, thus minimizing initial management tasks. While this default configuration may not be optimal, as some ASes might get rates that are too low, it nevertheless enhances communication availability for all legitimate traffic sources compared to no communication during DDoS attacks. To accommodate shifts in traffic patterns at both AS and ISD levels, router configurations can be adjusted in real-time. As for other SCION router configurations, this happens via gRPC [28]. Distinct configurations can be implemented based on the prevailing traffic conditions. Configurations can be created according to an AS' policies, or be learned based on ASes' and ISDs' demands during normal traffic situations, such that the corresponding guaranteed rates can later automatically be enforced during periods of active attacks. The infrastructure for traffic data collection already exists in most ISP networks, and, under normal conditions, links tend to be underutilized, which is likely to be reflected in the traffic history [13]. Therefore, a good configuration can be achieved by allocating rates slightly above their historical values, thus accounting for estimation inaccuracies and traffic burstiness. In Z-Lane, rate estimation is further simplified as only the small fraction of short-term intermediate-rate traffic needs to be considered (the majority of Internet traffic is caused by video streaming [58]). Numerous methods exist for addressing the challenge of traffic forecasting [23], which is therefore not the focus of our work. Suitable methods may for example include prediction models utilizing simple moving averages or more sophisticated artificial neural networks. Traffic forecasting enables operators to tailor configurations to their networks' specific requirements, minimize management overhead, and allow for automatic configuration adjustments over time. Consequently, increased management overhead only arises if the network operator chooses to intervene manually.

Configuring bandwidth on a per-ISD basis not only reduces memory overhead for border routers but also simplifies their configuration, as there is a significantly lower number of sources to monitor. When ASes are added or removed within a router's ISD, the configuration needs to be updated to reflect those changes. An AS can achieve this by consulting its DRKey service about the ASes that have recently requested or renewed a shared symmetric key, as those keys are a prerequisite for any source to send traffic using Z-Lane. Topology changes in other ISDs do not necessitate the addition or removal of new configuration entries; only the ISD's guaranteed rate may require occasional updates.

5 Analysis

We assess and compare the protective capabilities of Z-Lane routers against volumetric DDoS attacks (requirement R1) and investigate the effects of partial deployment and misconfigured per-AS and per-ISD rates.

5.1 Definitions

In our analysis, we only consider queues G and A, i.e., assuming a worst-case scenario where all other queues are fully utilized. We define our system at the granularity of border router interfaces, where an interface Y of router R has egress capacity $\mathrm{c}^{(\mathrm{R,Y})}$ (for traffic leaving the router). Traffic inside a router is described with variable f, where $\mathrm{f}_{(\mathrm{O,C})}^{(\mathrm{R,X,Y})}$ refers to traffic of class C from origin O (an AS or an ISD) that arrives at interface X and is to be forwarded to interface Y of border router R. There are two traffic classes C can assume:

- α: Comprises AS-level authenticated traffic without QoS field set as well as host-level authenticated traffic.
- γ: AS-level authenticated traffic with the QoS field set.

A router always tries to enqueue packets of class α at queue A. A packet of class γ is added to queue G if the router has configured a guaranteed rate for the packet's origin AS or ISD and if the packet does not lead to an overuse of that rate; otherwise the router tries to enqueue the packet at A (Figure 2). To describe traffic aggregates, we use the "·" notation. Traffic of class C arriving at router R that is to be forwarded to interface Y, for example, is written as $\mathrm{f}_{(\cdot,\mathrm{C})}^{(\mathrm{R},\cdot,\mathrm{Y})}$, which is an aggregate over all traffic origins and all ingress interfaces of router R. Finally, $\mathcal{A}(\mathrm{R})$ is a function returning a router's AS, and $\mathcal{D}(\mathrm{S})$ returns an AS' ISD. In the context of Z-Lane, $\mathrm{r}_{(\mathrm{O})}^{(\mathrm{R,\,Y})}$ is the egress bandwidth exclusively reserved to origin O at interface Y of border router R, i.e., R reserves a rate $\mathrm{r}_{(\mathrm{I})}^{(\mathrm{R,\,Y})}$ to ISD I if $\mathrm{I} \neq \mathcal{D}(\mathcal{A}(\mathrm{R}))$, and a rate $\mathrm{r}_{(\mathrm{S})}^{(\mathrm{R,\,Y})}$ if $\mathcal{D}(\mathrm{S}) = \mathcal{D}(\mathcal{A}(\mathrm{R})))$. If no rate is configured for some origin O, this is captured in our model by a value of zero for $\mathrm{r}_{(\mathrm{O})}^{(\mathrm{R,\,Y})}$.

5.2 Rates: ISD-Internal and ISD-External

Traffic from a certain origin O of class γ that is guaranteed to be forwarded at queue G can be computed as follows:

$$\mathrm{g}_{(\mathrm{O})}^{(\mathrm{R,\,Y})} := \min(\mathrm{f}_{(\mathrm{O},\gamma)}^{(\mathrm{R},\cdot,\mathrm{Y})},\ \mathrm{r}_{(\mathrm{O})}^{(\mathrm{R,\,Y})})$$

We refer to the guaranteed bandwidth for origin O that is potentially unused as follows:

$$\mathrm{u}_{(\mathrm{O})}^{(\mathrm{R,\,Y})} := \mathrm{r}_{(\mathrm{O})}^{(\mathrm{R,\,Y})} - \mathrm{g}_{(\mathrm{O})}^{(\mathrm{R,\,Y})}$$

Also, $\mathrm{u}_{(\cdot)}^{(\mathrm{R,\,Y})}$ is the sum of unused bandwidths over all possible AS- or ISD origins, respectively, or simply the unused bandwidth of $\mathrm{c}^{(\mathrm{R,Y})} \times \mu_G$. We denote any communication that exceeds the corresponding rate limit as overuse traffic:

$$\mathrm{o}_{(\mathrm{O})}^{(\mathrm{R,\,Y})} := \max\{\mathrm{f}_{(\mathrm{O},\gamma)}^{(\mathrm{R},\cdot,\mathrm{Y})} - \mathrm{r}_{(\mathrm{O})}^{(\mathrm{R,\,Y})},\ 0\}$$

Again, $\mathrm{o}_{(\cdot)}^{(\mathrm{R,\,Y})}$ refers to the total overuse traffic of all origin ASes and ISDs. As we assume that the queues R, N, and U are fully utilized, the leftover egress capacity $\mathrm{l}^{(\mathrm{R},\mathrm{Y})}$, which can be used for overuse traffic from G and for traffic of class α, is the lower bound of A plus the underutilized capacities at G:

$$\mathrm{l}^{(\mathrm{R},\mathrm{Y})} = (\mathrm{c}^{(\mathrm{R},\mathrm{Y})} \times \mu_A) + \mathrm{u}^{(\mathrm{R},\,\mathrm{Y})}_{(\cdot)}$$

In case there is no congestion, i.e., if $\mathrm{o}^{(\mathrm{R},\mathrm{Y})} + \mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\cdot,\alpha)} \leq \mathrm{l}^{(\mathrm{R},\mathrm{Y})}$, the best-effort traffic $\mathrm{b}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{O})}$ that origin O can send at queue A is equal to its overuse traffic $\mathrm{o}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{O})}$ plus its traffic $\mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{O},\alpha)}$ of class α. In case of congestion, the leftover capacity $\mathrm{l}^{(\mathrm{R},\mathrm{Y})}$ is shared with all other origin ISDs and ASes.

$$\mathrm{b}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{O})} := \min\{\frac{\mathrm{l}^{(\mathrm{R},\mathrm{Y})}}{\mathrm{o}^{(\mathrm{R},\mathrm{Y})} + \mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\cdot,\alpha)}},\ 1\} \times (\mathrm{o}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{O})} + \mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{O},\alpha)})$$

We can now calculate the rate at which a router R forwards traffic originating from AS S at interface Y. This calculation depends on whether R is inside or outside the ISD of AS S.

ISD-Internal. If $\mathcal{D}(S) = \mathcal{D}(\mathcal{A}(\mathrm{R}))$, the total throughput $\mathrm{t}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})}$ of AS S can be computed as the sum of traffic forwarded at queue G and A together:

$$\mathrm{t}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} := \mathrm{g}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} + \mathrm{b}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} \tag{1}$$

ISD-External. If $\mathcal{D}(S) \neq \mathcal{D}(\mathcal{A}(\mathrm{R}))$, with $\mathrm{I} := \mathcal{D}(S)$, we can compute both the traffic volume $\widehat{\mathrm{g}}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{S})}$ of AS S forwarded through queue G based on the guaranteed ISD rate, as well as the traffic volume $\widehat{\mathrm{b}}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{S})}$ of AS S forwarded through queue A based on its ISD's best-effort share.

$$\widehat{\mathrm{g}}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{S})} := \mathrm{g}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{I})} \times \frac{\mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{S},\gamma)}}{\mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{I},\gamma)}},$$

$$\widehat{\mathrm{b}}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{S})} := \mathrm{b}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{I})} \times (\frac{(\mathrm{o}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{I})} \times \frac{\mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{S},\gamma)}}{\mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{I},\gamma)}}) + \mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{S},\alpha)}}{\mathrm{o}^{(\mathrm{R},\,\mathrm{Y})}_{(\mathrm{I})} + \mathrm{f}^{(\mathrm{R},\cdot,\mathrm{Y})}_{(\mathrm{I},\alpha)}})$$

The total throughput $\mathrm{t}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})}$ of AS S can again be computed as the sum of traffic forwarded at queues G and A together:

$$\mathrm{t}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} := \widehat{\mathrm{g}}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} + \widehat{\mathrm{b}}^{(\mathrm{R},\mathrm{Y})}_{(\mathrm{S})} \tag{2}$$

5.3 DDoS Attack Analysis

Based on the theory explored so far, we now analyze Z-Lane's resistance against volumetric DDoS attacks. For this, we consider a specific inter-domain path $\mathrm{p} = [(\mathrm{R}_1, \mathrm{X}_1, \mathrm{Y}_1), \ldots, (\mathrm{R}_\ell, \mathrm{X}_\ell, \mathrm{Y}_\ell)]$, i.e., a list of border routers and their interfaces, over which an AS S sends traffic. To simplify our analysis, we assume that no other inter-domain path used by S intersects any egress interface of p:

$$\underset{1\leq n\leq \ell}{\forall n:}\quad \mathrm{f}^{(\mathrm{R}_n,\mathrm{X}_n,\mathrm{Y}_n)}_{(\mathrm{S},\gamma)} = \mathrm{f}^{(\mathrm{R}_n,\cdot,\mathrm{Y}_n)}_{(\mathrm{S},\gamma)}$$

In accordance with SCION's path segment combination mechanism, the first j on-path routers are inside the same ISD as S, while the last (ℓ-j) routers are in different ISDs (Figure 1):

$$\underset{1\leq n\leq j}{\forall n:}\quad \mathcal{D}(S) = \mathcal{D}(\mathcal{A}(\mathrm{R}_n)), \qquad \underset{j<n\leq \ell}{\forall n:}\quad \mathcal{D}(S) \neq \mathcal{D}(\mathcal{A}(\mathrm{R}_n))$$

We further assume that for every on-path router the egress interface capacity is C, and that the total traffic from S' ISD, without the traffic from S itself, is T:

$$\underset{1\leq n\leq \ell}{\forall n:}\ \mathrm{c}^{(\mathrm{R}_n,\mathrm{Y}_n)} = \mathrm{C}, \qquad \underset{j<n\leq \ell}{\forall n:}\ \mathrm{f}^{(\mathrm{R}_n,\cdot,\mathrm{Y}_n)}_{(\mathrm{I},\gamma)} - \mathrm{f}^{(\mathrm{R}_n,\cdot,\mathrm{Y}_n)}_{(\mathrm{S},\gamma)} \quad = \mathrm{T}$$

We use the example scheduling bounds from Table 2.

Z-Lane. We assume that all communication originating from S is sent over the Z-Lane gateway adding a QoS flag to every packet, and that on-path routers have the same guaranteed rates configured for S or $\mathcal{D}$(S), respectively:

$$\underset{1\leq n\leq j}{\forall n:}\quad \mathrm{r}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})} = \mathrm{r}_{(\mathrm{S})}, \qquad \underset{j+1\leq n\leq \ell}{\forall n:}\quad \mathrm{r}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{I})} = \mathrm{r}_{(\mathrm{I})}$$

We consider a scenario where at every on-path router the guaranteed rates of ASes and ISDs are always fully utilized, with the exception of the rates of S and $\mathcal{D}$(S), which depend on the throughput achieved at the previous entity:

$$\mathrm{f}^{(\mathrm{R}_1,\cdot,\mathrm{Y}_1)}_{(\mathrm{S},\gamma)} = \mathrm{F}, \qquad \mathrm{f}^{(\mathrm{R}_{n+1},\cdot,\mathrm{Y}_{n+1})}_{(\mathrm{S},\gamma)} = \mathrm{t}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})}$$

For the evaluation, we vary the transmission rate F of S and measure the achieved throughput $\mathrm{t}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})}$ for each Y_n. We recall that the computation of $\mathrm{t}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})}$ differs for routers inside and outside $\mathcal{D}$(S) (Equations (1) and (2)).

FIFO Router. As a first baseline, we compute the achieved throughput to routers with FIFO queuing only, meaning that all traffic goes into one single egress queue with capacity $\mathrm{c}^{(\mathrm{R},\mathrm{Y})} \times (\mu_A + \mu_G)$. Therefore, in case of congestion, the throughput that AS S gets on some on-path router R_n decreases with the total traffic directed at interface Y_n:

$$\mathrm{t}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})} = \min\{\frac{\mathrm{c}^{(\mathrm{R}_n,\mathrm{Y}_n)} \times (\mu_A + \mu_G)}{\mathrm{f}^{(\mathrm{R}_n,\cdot,\mathrm{Y}_n)}_{(\cdot,\cdot)}},\ 1\} \times \mathrm{f}^{(\mathrm{R}_n,\cdot,\mathrm{Y}_n)}_{(\mathrm{S},\cdot)}$$

We work under the same assumptions as with Z-Lane, i.e., $(\mathrm{c}^{(\mathrm{R},\mathrm{Y})} \times \mu_G) - \mathrm{r}_{(\mathrm{S})}$ is fully used by other ASes or ISDs, and all their additional traffic is considered DDoS.

Per-flow Fairness. As a second baseline, we also evaluate a router enforcing per-flow fair rates. In this evaluation, the attacker is aware of this strategy and therefore creates twice the number of flows per traffic volume than the benign parties.

Local Isolation. Our third baseline provides bandwidth isolation exclusively based on AS-local information such as ingress-egress interfaces, representing mechanisms such as RCS [8] and PSP [13].

Results. We instantiate C, T, $\mathrm{r}_{(\mathrm{S})}$, and $\mathrm{r}_{(\mathrm{I})}$ with 40, 3, 1, and 4 Gbps, respectively. Figure 4 shows the source AS' throughput $\mathrm{t}^{(\mathrm{R}_n,\mathrm{Y}_n)}_{(\mathrm{S})}$ at each on-path border router $\mathrm{n} \in \{1, \ldots, 8\}$ when

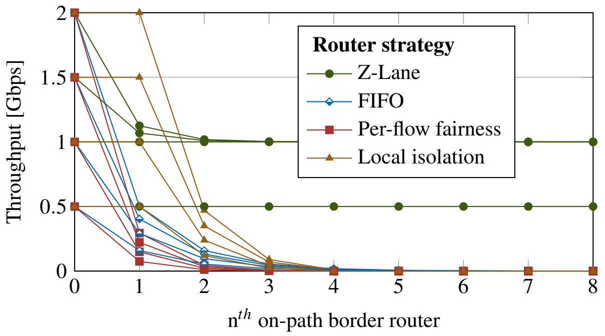

Figure 4: The source AS' achieved throughput at on-path routers, each under an attack twice as large as the egress capacity (traffic intensity X = 2), for different source transmission rates ($F \in \{\frac{1}{2}, 1, \frac{3}{2}, 2\}$ Gbps) and router implementations.

under a DDoS attack of D = 40 Gbps, i.e., with a traffic intensity of $X = \frac{C+D}{C} = 2$.[5] For all values of the source AS' traffic rate F, the baseline results show an exponential decrease in throughput along the path. While local isolation can fully protect traffic at the first hop due to an interface-pair allocation unaffected by DDoS, it subsequently gets merged with attack flows, therefore sharing the same future allocations. With Z-Lane, for F > 1 Gbps, the throughput also decreases exponentially, but it never drops below 1 Gbps. Even for only three attacked border routers, Z-Lane provides up to 30 times higher throughput than the baseline. For $F \leq 1$ Gbps there is no packet loss, the achieved throughput is always equal to F. Therefore, a misconfigured rate that has been set too high is ideal for the source AS as it protects all of its Z-Lane traffic, however, the unused guaranteed bandwidth could be of better use when assigned to a different AS or ISD. In contrast, a rate set too low means that the AS and its end hosts do not get all their desired traffic protected from DDoS attacks, meaning that the end hosts' applications must lower their rates until their sum is smaller than their AS' guaranteed rate.

The results of a partial Z-Lane deployment, where some on-path ASes either do not support Z-Lane or have not configured a rate for the source AS, can also be deduced from Figure 4. For $F \leq 1$ Gbps, the outcomes align with the baseline experiments, where the x-axis signifies the number of ASes not providing guaranteed rates.

Congestion Control. We anticipate the FIFO results presented here to be even lower in reality. In the absence of attackers, when congestion occurs, end-to-end congestion control mechanisms, such as those employed by TCP, compel traffic sources to reduce their transmission rates to mitigate the situation. Under a large DDoS attack, however, attackers do not back off, resulting in invariably congested links. This in turn causes a vicious cycle of benign sources over and over reducing their transmission rate, i.e., actively lowering F. It has been shown that only very little congestion is enough for legitimate flows to reduce their rate [47, 51]. In Z-Lane, as soon as $F \leq 1$ Gbps, this cycle is interrupted and communication is still possible at 1 Gbps.

5.4 Security Analysis

Strategy. Unlike many other proposed solutions, Z-Lane does *not attempt to detect or actively mitigate* volumetric DDoS attacks, as this is impossible in our threat model, where attack traffic might be indistinguishable from benign communication. Instead, Z-Lane looks at the problem from a different angle, focusing on how to *protect benign communication* from getting dropped in transit. For an attacker-free forwarding path, Z-Lane's per-AS and per-ISD rates can effectively contribute to sustaining legitimate communication even in the face of the latest and most prevalent DDoS attacks, including botnet-size variants such as those seen in Mirai, or other elaborate techniques such as pulse-wave attacks.

Trust. In general, our threat model permits any combination of off-path ASes and ISDs (but not on-path ones) to be compromised. Due to the per-AS and per-ISD bandwidth isolation enforced by Z-Lane, the impact of a compromised off-path AS or ISD generating vast amounts of traffic is limited. Still, because of ISD-level (as opposed to AS-level) bandwidth isolation at routers outside an AS' ISD, the AS generally needs to trust other AS operators in its ISD. This trust is usually warranted due to ISDs being constructed based on a uniform trust environment or a common jurisdiction (Section 2). We highlight that such trust is not required regarding other ASes' end hosts. Even if a malicious AS would launch a flooding attack, other ASes inside its ISD, in particular also core ASes (which are at the boundary to other ISDs), can use Z-Lane's AS-level bandwidth isolation mechanism to prevent traffic floods from even reaching other ISDs.

Configuration. The configurable end host rates at the Z-Lane gateway protect benign end hosts from malicious end hosts in the same network trying to deplete their AS' guaranteed rates, and the configurable per-AS and per-ISD rates at border routers prevent Sybil attacks, where an attacker would create many ASes in an attempt to get a higher share of the routers' egress capacity. In today's Internet, large-scale volumetric DDoS attack have the potential to disrupt all communication over a certain link. With Z-Lane, even if rates were not optimally configured, every AS still strictly benefits compared to a situation without Z-Lane in place.

Spoofing. Successful spoofing of source addresses enables the exploitation of per-source rate limiting, potentially resulting in the disruption of traffic from a benign source. Therefore, in a Z-Lane router, if one AS can spoof traffic from another AS, it can exhaust its guaranteed rate. Similarly, at the Z-Lane gateway, if an end host can spoof traffic from another end host, it can deplete its allocated bandwidth. These observa-

[5] The average Internet path length is five AS-level hops [71], translating to five to eight on-path border routers, depending on the path crossing one or two border routers at each transit AS.

tions motivate our reliance on EPIC's source authentication, which enables us to effectively mitigate such attacks.

6 Implementation and Evaluation

We demonstrate Z-Lane's integration into the SCION ecosystem, its incremental deployability, and its efficiency by implementing and evaluating it on a global testbed and as high-performance application.

6.1 SCIONLab Implementation

We implement a prototype version of Z-Lane's configurable scheduling mechanism in SCIONLab [2, 41], a global testbed that allows research and experimentation with SCION. The main purpose of this evaluation is to show that implementing and integrating Z-Lane in SCION is feasible, does not break other components, and can be performed incrementally.[6] As SCIONLab ASes are controlled by many different individuals and organizations and as we do not have access to their infrastructure, we add our own ASes to SCIONLab. To demonstrate fully protected forwarding, some of our ASes are directly connected to each other, and to demonstrate incremental deployment, others are connected to ASes of other entities around the globe. We obtained permission from the SCIONLab operators to conduct our measurements. During the three months for which our prototype was running, we have not observed any anomalous events. Moreover, we measured traffic statistics of a source AS for which we configured a rate of 10 Mbps at on-path routers that we congested with communication from other ASes, where the source AS always generated traffic below the configured rate. With normal SCION traffic we observed high latency and packet loss; with Z-Lane traffic on the other hand, all packets were successfully forwarded to the destination without significant change in latency. Unfortunately, this setup is not sufficient to demonstrate Z-Lane's defense against DDoS attacks exceeding a few hundred Mbps, as (i) our border router written in Go is not optimized for high performance and (ii) the majority of SCIONLab's inter-domain links are IP overlays, and might thus actually share the same underlying forwarding infrastructure. Also, the SCIONLab topology is not large enough to simulate Internet-scale behavior of Z-Lane. For this reason, we also implement and evaluate high-speed prototypes of Z-Lane's components.

6.2 High-Speed Implementation

We implement high-speed versions of the Z-Lane gateway and the Z-Laneborder router using Intel's Data Plane Development Kit (DPDK) [21].[7] We conduct performance measurements by executing both components one after the other (never at the same time) on a commodity machine with an Intel Xeon processor running at 2.1 GHz, while generating and monitoring traffic on a Spirent SPT-N4U. Four 40 Gbps bidirectional Ethernet links provide connectivity between the two machines. Both the gateway and the border router read packets from the corresponding ports, process them, and send them back to the Spirent machine. We implement the additional MAC computation introduced in Z-Lane (Appendix C) with AES in CBC mode, where we speed up the block cipher computation by taking advantage of Intel's AES-NI [30] hardware support. To avoid false sharing, we cache-align data structures like the token buckets.

Table 3: Mean gateway processing time for packets arriving from 32 000 different hosts. Highlighted tasks are only executed when the packet does not exceed a host's rate limit.

Task	Time [ns]
Parse and verify the received Z-Lane packet	62
Look up the token bucket (TB)for the source end host	42
Check whether the packet exceeds the TB rate	101
Subtotal	205
Set the QoS flag in the packet header	12
Look up the AS-level key	$38 \times \ell$
Compute the new/updated HVF	$61 \times \ell$
Overwrite the old with the new HVF	$9 \times \ell$
Total	$217 + 108 \times \ell$

Z-Lane Gateway. End host rate limiting at the gateway is implemented with a hash table mapping end host addresses to token buckets. We evaluate the gateway's throughput and per-packet processing time in an AS with 32 000 active end hosts. We distinguish between packets that are compliant with the corresponding end host's rate and packets that lead to an overuse of this rate. Overuse packets can be filtered early, i.e., without the gateway having to re-compute their HVFs.

A fine-grained analysis of the gateway's packet processing times is shown in Table 3. While filtering an overuse packet only takes 205 ns, packets that are rate-compliant pose an additional overhead of 13 ns to set the QoS flag plus 108 ns per on-path AS to update the corresponding HVF. Those numbers are negligible compared to end-to-end latencies that are typically on the order of milliseconds (requirement R4).

Figure 5 shows the gateway's forwarding performance for traffic sent over a path traversing five ASes.[8] Processing exclusively overuse packets compared to only processing rate compliant packets requires approximately four times less cores to achieve the same forwarding rate, which is consistent with the packet processing time results. At any point in time, the actual forwarding speed of the gateway is somewhere between the

[6] Z-Lane can be incrementally deployed where EPIC is implemented with separate MACs for path authorization and source authentication and where packets failing source authentication are forwarded with best effort.

[7] We chose DPDK because it (i) enables fast prototyping of data plane protocols, (ii) is supported on many devices [22], and (iii) is used by manufacturers of SCION border routers [5]. Still, any production-ready implementation of Z-Lane does not necessarily need to be based on DPDK.

[8] The average Internet path length is five AS-level hops [71].

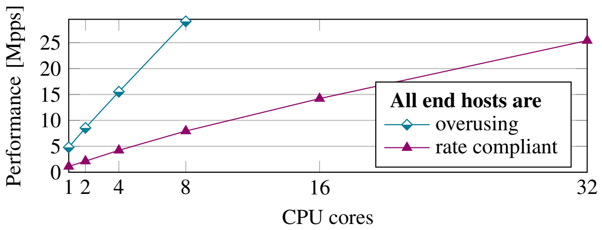

Figure 5: Gateway packet forwarding performance for different number of CPU cores and (i) overuse traffic only, compared to (ii) only traffic complying with the given rate.

evaluated overuse-only and the evaluated rate-compliant-only performance and depends on the end host traffic volumes and the configured rates. With a worst-case performance of 25.4 Mpps when using 32 cores,[9] we consider the gateway viable for real-world deployment scenarios. If more performance is needed, additional gateways can be added, where all traffic from a certain end host is forwarded to the same gateway instance to avoid synchronizing token bucket state.

Z-Lane Border Router. In the router, cores read and classify packets from their assigned ports and add them to queues, implemented by means of lock-free ring buffers, allocated for the corresponding egress port. A dedicated egress core reads the packets from its queues and schedules them according to the example capacity bounds in Table 2.For the queues R and N, we enforce their capacity upper bound through dedicated token buckets. Similarly, to implement the per-AS and per-ISD rate limiting, we use hash tables to map AS and ISD identifiers to token buckets enforcing those rates. For the evaluation of the border router, the Spirent machine generates up to 160 Gbps of traffic from a configurable mix of different traffic classes with an average payload size of ∼750 B, which is motivated by COLIBRI incentivizing the use long payloads to maximize the senders' goodput, while other traffic is expected to follow a packet size distribution with lower median [35]. Authenticated traffic with a QoS marker originates from 1000 ASes of the router's ISD and from 1000 other ISDs, which corresponds to an Internet topology of approximately 10^6 ASes. We evaluate the worst-case scenario, where all traffic is directed towards the same egress link.

To demonstrate the correct enforcement of the queues' lower and upper bounds, we generate up to 40 Gbps of different traffic mixes. The restriction to 40 Gbps in this evaluation ensures that packets are not dropped due to congestion, but from the enforcement of the upper bounds only. The results are shown in Figure 6. We observe that neither the 20 Gbps upper bound of R nor the 4 Gbps upper bound of N are exceeded at egress. Furthermore, traffic for G, A, and U are not upper bounded and can use any remaining capacity (traffic for G can overuse G's upper bound because overuse packets are enqueued at A, which is not upper bounded).

To show the system's resistance against volumetric DDoS attacks, we track one particular AS X for which the border router is configured to allow a guaranteed throughput of 928 Mbps. We generate 159.072 Gbps of attack traffic from the different classes, where we overuse the upper bounds from R, N, and all the ISD and AS guarantees, with the exception of AS X. The results are shown in Table 4. We observe that the different traffic classes are correctly enforced. As all ASes apart from AS X are overusing their guarantees, where the overuse traffic goes to A, the throughput of authenticated traffic without the QoS marker set is relatively low, as it competes for bandwidth against this overuse traffic. As a desirable side effect, traffic sent over G, R, and N is forwarded with minimal latency and jitter, which results from their upper bounded capacity preventing those queues to grow.

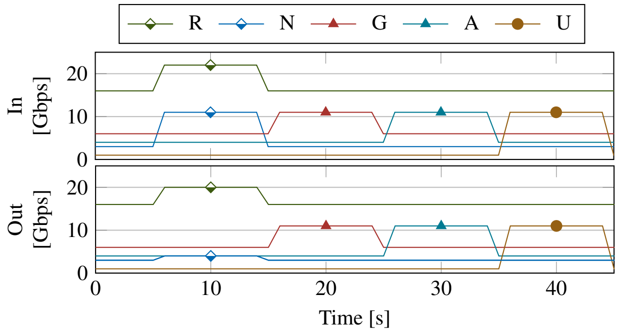

Figure 6: Router scheduling output in a congestion-free setting, i.e., the total ingress traffic rate never exceeds the egress capacity of 40 Gbps. The upper bounds of R (20 Gbps) and N (4 Gbps) are correctly enforced, and traffic to G, A, and U can use free capacity as desired.

Table 4: Border router DDoS resistance measurement results in Gbps. Traffic from AS X (part of G) is successfully forwarded at the configured rate of 928 Mbps.

	R	N	G	AS X	A	U	Total
In	64.0	12.0	37.0	0.928	30.0	17.0	160.0
Out	19.99	3.99	10.56	0.927	3.37	1.98	39.89

The main challenge in terms of scalability is the concurrent access of different cores to the same lock-protected per-AS or per-ISD token bucket for packets destined to the same egress. This can potentially lead to increased lock contention for routers with a high number of ports. To solve this problem, the router can be replaced by multiple ones, each responsible for a subset of the original AS interfaces. Another solution is to reduce egress contention by performing additional rate limiting at ingress, where the ingress rate enforced for an AS or ISDs is the maximum of its rates configured at the egresses. This idea can be further extended to per-egress hierarchical rate limiting, where additional intermediate cores receive rate-limited packets destined to a specific egress from several other child cores, rate limit them again, and forward the rate-compliant packets to the next core in the hierarchy until they arrive at the top of the hierarchy. Alternatively, rate limiting

[9] For 500 B packets this means ∼100 Gbps, for 800 B packets ∼160 Gbps.

can also be performed in the NIC [56]. To provide better support for Z-Lane also on memory constrained devices, we describe in Appendix D an optimized rate limiter reducing the memory overhead compared to current state-of-the-art implementations by 60-86%. Z-Lane's deployment is further discussed in the following section.

7 Discussion

Lowering μ_{U}. An interesting configuration edge case is a value of zero for μ_U, i.e., for the router's lower scheduling bound for unauthenticated best-effort traffic (Table 2). This setting implies that in congestion-free periods unauthenticated traffic still gets through, but can starve completely during congestion. Even though the AS-level keys required for EPIC are initially requested based on unauthenticated best-effort, in case of congestion in `U` the keys can still be fetched using DoCile instead. Therefore, this setting does not introduce additional venues for DDoS. Instead, it provides benefits by (i) incentivizing the use of authenticated traffic and (ii) preventing the undesired situation where there is congestion in `A` but not in `U`, leading to an end host's authenticated best-effort traffic being dropped, while its unauthenticated best-effort traffic is still forwarded successfully. We emphasize that setting μ_{U} to zero is semantically not the same as removing the queue `U`.

Deployment Incentives and Future Work. Deploying Z-Lane is facilitated in several ways. The per-AS and per-ISD rates for example are defined independently by each AS without inter-domain coordination. Also, configuration complexity is low and Z-Lane can be incrementally deployed. For transit ASes, deploying Z-Lane means updating the software running on their SCION border routers, while source ASes have to run a Z-Lane gateway. As Z-Lane runs over existing public Internet infrastructure, it incurs low costs (requirement R5), while achieving availability guarantees similar to significantly more expensive leased lines or private backbones (Section 8). With Z-Lane, an ISP can maintain low-latency Internet connectivity despite DDoS attacks and thus offer better service and attract more customers, providing incentives for early adopters. Every source AS benefits from a transit AS allocating a guaranteed rate, even if that rate is lower than its demand. This contrasts complete communication outage due to current routers with mostly FIFO queuing and drop-tail mechanisms failing to protect against DDoS attacks (Section 5.3). Directions for future work include the implementation of Z-Lane on different hardware, the design of a protocol to automatically learn traffic profiles for router configurations, and a formal verification of Z-Lane's ecosystem, i.e., in combination with COLIBRI, DRKey, and DoCile.

8 Related Work

Table 5 shows that, compared to existing systems, Z-Lane excels in protecting short-lived intermediate-rate traffic.

One widely employed mechanism is network scrubbing, where traffic is redirected to a server cluster for filtering out malicious packets. The setup process is time-consuming, operationally complex, and may necessitate changes to the network architecture, and, once in place, it creates vendor dependency (vendor lock-in) and becomes a single point of failure, while introducing additional latency also for benign traffic. IP address spoofing can lead to traffic from benign sources being attributed as malicious, leading to collateral damage. Other solutions like CDNs reduce the attack surface by shortening communication paths to replicated services but can only protect certain path segments. While they can handle multiple terabits per second, they do not provide comprehensive DDoS defense against nation-state attackers and can be prohibitively expensive during attacks. Solutions like overprovisioning network capacities only shift the problem rather than addressing it fundamentally. Overprovisioning is financially expensive, and available capacity could be used more effectively. While leased lines and private backbones offer robust communication guarantees, they are not scalable for protecting billions of end hosts and come with significant financial costs.

Within academia, more economical solutions have been explored, primarily focusing on implementing DDoS defenses at routers. Unfortunately, the effectiveness of those solutions diminishes when faced with adversaries capable of spoofing IP packets, rendering them inadequate for offering fundamental protection against attacks. These proposed solutions encompass various approaches, ranging from history-based IP address filtering at edge routers [54] to fairness strategies for traffic flows [63, 76], as well as strategies for mitigating DDoS attack on programmable switches [45, 77]. An example is Pushback [46], which aims to identify flow aggregates responsible for congestion and subsequently imposes rate limits on these aggregates at upstream routers. However, it cannot provide instant protection for benign traffic against DDoS attacks nor defend against botnets. Even very recent proposals follow such a reactive pattern, i.e., they try to detect, classify, and then mitigate attacks [3]; in contrast, Z-Lane works proactively and can thus protect against threats instantaneously. To achieve flexibility regarding the deployment of congestion control algorithms (CCAs), RCS [8] enforces isolation at egress without necessitating inter-domain agreements. Nevertheless, its performance rates exhibit exponential decreases in path length when on-path routers are subjected to attacks. Similarly, PSP allocates bandwidth between interface-pairs based on historical traffic measurements [13]. Bandwidth reservation systems [7, 27, 73, 75] run an expensive setup process, which is necessary to make sure all on-path ASes agree on the allocated bandwidth and to fetch the necessary cryptographic tokens ("capabilities") required to send reservation traffic. Also, most such systems do not protect traffic on the return path. Appendix H elaborates on related work regarding source authentication systems, highlighting why only EPIC

Table 5: Various systems from industry and academia and their suitability to protect short-lived intermediate-rate traffic.

	"Zero-setup"	Low management overhead	General end user protection	Low financial costs	Fundamental guarantees	Arbitrarily high DDoS	No impact on latency	Protect return traffic	Resistant to IP spoofing	No collateral damage
Rerouting and Scrubbing	-	-	●	-	-	-	-	-	-	-
CDN	●	-	-	-	-	-	○	○	-	○
Overprovisioning	●	-	●	-	-	-	●	●	-	-
Leased lines	●	-	-	-	●	●	●	●	●	●
Private backbone	●	-	-	-	●	●	●	●	●	●
History-based filtering	●	●	●	●	○	-	●	○	-	-
Per-flow fairness	●	●	●	●	-	-	●	●	-	-
Pushback	-	○	●	●	-	-	○	●	-	-
Bandwidth reservations	-	●	●	●	●	●	●	○	●	●
Z-Lane	●	●	●	●	●	●	●	●	●	●

●: property achieved, -: property not achieved, ○: property partially achieved

can provide the necessary foundation for meaningful resource allocation at routers.

9 Conclusion

Protecting the availability of short-lived, intermediate-rate traffic of protocols like DNS, HTTP(S), or SSH is challenging. In-network volumetric DDoS defense solutions are often reactive, vulnerable to source address spoofing, fail to defend against legitimately looking attack traffic, cannot defend against attacks on longer paths, or significantly increase time to communication due to expensive defense setup procedures. We take a different look at the problem: letting every border router check the authenticity of every packet's source, Z-Lane can securely enforce AS-level bandwidth isolation at border routers, which is efficiently implemented through configurable forwarding lower bounds for (groups of) autonomous systems (ASes). Thus, every AS gets "zero-setup" forwarding guarantees that can be extended to its end hosts and that persist irrespective of off-path adversaries' attack patterns.

Z-Lane is both practical and cost-effective: traffic is sent over existing public Internet infrastructure, and deploying Z-Lane is simple and can be done incrementally, while it achieves properties similar to private or leased infrastructure. Z-Lane scales to Internet topologies consisting of hundreds of thousands of ASes, while only requiring five queues (two of them enable integration of related systems and are therefore optional) and a few kilobytes of memory per router interface.

Z-Lane provides a foundation for building exciting new systems on the public Internet, such as highly available industrial command-and-control or DNS systems, with strong communication guarantees despite volumetric DDoS attacks.

Acknowledgments

We extend our gratitude to Juan Angel García-Pardo, Francesco Da Dalt, and Marc Odermatt for their thorough review and constructive feedback on the manuscript, Fabio Streun for his assistance with the hardware setup, Giacomo Giuliari for his insightful feedback on the system, and Yih-Chun Hu for the enlightening discussions. We are also thankful to the anonymous reviewers for their valuable input and to our shepherd for guiding us through the final version of the paper. We gratefully acknowledge the support received from ETH Zurich, the Zurich Information Security and Privacy Center (ZISC), and armasuisse Science and Technology.

References

[1] Bidirectional Forwarding Detection. RFC 5880, 2010.

[2] The SCIONLab research network. https://www.scionlab.org/, 2024.

[3] Albert Gran Alcoz et al. Aggregate-based congestion control for pulse-wave ddos defense. SIGCOMM, 2022.

[4] Anapaya Systems. Products for industry. https://www.anapaya.net/products-for-industry, 2024.

[5] Anapaya Systems. SCION-internet and anapaya software. http://tinyurl.com/3v8hu43f, 2024.

[6] Anapaya Systems. SCION-Internet: The New Way To Connect. https://www.anapaya.net/scion-the-new-way-to-connect, 2024.

[7] Cristina Basescu et al. SIBRA: Scalable Internet Bandwidth Reservation Architecture. NDSS, 2016.

[8] L. Brown et al. On the future of congestion control for the public Internet. HotNets, 2020.

[9] Canonical. Ubuntu manpage - deficit round robin scheduler. https://tinyurl.com/4ss2dpe2, 2024.

[10] Canonical. Ubuntu manpage - enhanced transmission selection scheduler. https://tinyurl.com/47pmmzwe, 2024.

[11] Canonical. Ubuntu manpage - hierarchy token bucket scheduler. https://tinyurl.com/2jbhb79b, 2024.

[12] Catherine G. Manning. Technology readiness levels - NASA. http://tinyurl.com/5yh5ut3p, 2024.

[13] Jerry Chou et al. Proactive surge protection: a defense mechanism for bandwidth-based attacks. USENIX Security, 2008.

[14] Laurent Chuat et al. *The Complete Guide to SCION*. Springer International Publishing, 2022.

[15] Cloudflare. Cloudflare DDoS threat report for 2022 q4. https://tinyurl.com/25s57wdp, 2024.

[16] Cloudflare. Cloudflare DDoS threat report for 2023 q2. https://tinyurl.com/y899ftth, 2024.

[17] Cloudflare. Cloudflare mitigates record-breaking 71 million request-per-second DDoS attack. http://tinyurl.com/yd2zkmjv, 2024.

[18] Daryll Swer. How i set up my own autonomous system. http://tinyurl.com/sxbrt5ux, 2024.

[19] C. de Kater, N. Rustignoli, and A. Perrig. SCION overview. http://tinyurl.com/mwzbska5, 2024.

[20] A. Demers et al. Analysis and simulation of a fair queueing algorithm. *SIGCOMM CCR*, 1989.

[21] DPDK Project. Data Plane Development Kit. https://dpdk.org, 2024.

[22] DPDK Project. DPDK: Supported hardware. https://core.dpdk.org/supported/, 2024.

[23] G. O. Ferreira et al. Forecasting network traffic: A survey and tutorial with open-source comparative evaluation. *IEEE Access*, 2023.

[24] S. Floyd et al. Random early detection gateways for congestion avoidance. *IEEE/ACM ToN*, 1993.

[25] Songtao Fu et al. MASK: Practical source and path verification based on Multi-AS-Key. IWQoS, 2021.

[26] Github Inc. Open vSwitch rate limiter in c. http://tinyurl.com/3vk33dyw, 2024.

[27] Giacomo Giuliari et al. Colibri: A cooperative lightweight inter-domain bandwidth-reservation infrastructure. CoNEXT, 2021.

[28] Google. gRPC. https://grpc.io/, 2024.

[29] Google Open Source. Golang rate limiter. https://tinyurl.com/yun2k6h5, 2024.

[30] Shay Gueron. Intel Advanced Encryption Standard (AES) new instructions set. Technical report, Intel, 2010.

[31] Carl A. Gunter et al. Dos protection for reliably authenticated broadcast. NDSS, 2004.

[32] GÉANT. SCION Access for Universities and Research Institutes. https://rb.gy/xcnrpk, 2024.

[33] Anxiao He et al. Hummingbird: Dynamic path validation with hidden equal-probability sampling. *TIFS*, 2023.

[34] Yongchao He et al. Scalable on-switch rate limiters for the cloud. IEEE INFOCOM, 2021.

[35] Wolfgang John et al. Analysis of internet backbone traffic and header anomalies observed. IMC, 2007.

[36] Min Suk Kang et al. The crossfire attack. S&P, 2013.

[37] Chris Karlof et al. Distillation codes and applications to dos resistant multicast authentication. NDSS, 2004.

[38] Tiffany Hyun-Jin Kim et al. Lightweight source authentication and path validation. ACM SIGCOMM, 2014.

[39] Maciej Korczyński and Yevheniya Nosyk. *Source Address Validation*. 2019.

[40] C. Krähenbühl et al. Deployment and scalability of an inter-domain multi-path routing infrastructure. CoNEXT, 2021.

[41] Jonghoon Kwon et al. SCIONLAB: A next-generation internet testbed. ICNP, 2020.

[42] Markus Legner et al. EPIC: Every packet is checked in the data plane of a path-aware Internet. USENIX Security, 2020.

[43] Guanyu Li et al. Nethcf: Enabling line-rate and adaptive spoofed ip traffic filtering. ICNP, 2019.

[44] X. Liu, A. Li, X. Yang, and D. Wetherall. Passport: Secure and adoptable source authentication. 2008.

[45] Zaoxing Liu et al. Jaqen: A high-performance switch-native approach for detecting and mitigating volumetric ddos attacks with programmable switches. USENIX Security, 2021.

[46] Ratul Mahajan et al. Controlling high bandwidth aggregates in the network. *SIGCOMM CCR*, 2002.

[47] Matthew Mathis et al. The macroscopic behavior of the tcp congestion avoidance algorithm. *CCR*, 1997.

[48] Matthew Prince. The DDoS that knocked spamhaus offline. http://tinyurl.com/s2hjaa4u, 2024.

[49] D Medhi and K Ramasamy. *Network Routing: Algorithms, Protocols, and Architectures*. 2007.

[50] J. Naous et al. Verifying and enforcing network paths with ICING. CoNEXT, 2011.

[51] Neal Cardwell. BBR v2: A model-based congestion control. http://tinyurl.com/25khc6et, 2024.

[52] NETSCOUT. DDoS Threat Intelligence Report. https://www.netscout.com/threatreport/, 2024.

[53] A.K. Parekh et al. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM ToN*, 1993.

[54] Tao Peng et al. Protection from distributed denial of service attacks using history-based ip filtering. ICC, 2003.

[55] Maxime Piraux et al. Revealing the evolution of a cloud provider through its network weather map. IMC, 2022.

[56] Sivasankar Radhakrishnan et al. SENIC: Scalable NIC for end-host rate limiting. NSDI, 2014.

[57] Benjamin Rothenberger et al. PISKES: Pragmatic Internet-scale key-establishment system. ASIACCS, 2020.

[58] S. Gutelle. In 2022, 65% of all internet traffic came from video sites. http://tinyurl.com/f93xcm8e, 2024.

[59] Simon Scherrer et al. Low-rate overuse flow tracer (LOFT): An efficient and scalable algorithm for detecting overuse flows. SRDS, 2021.

[60] SCION Project. SCION open-source implementation. https://github.com/scionproto/scion, 2024.

[61] Micah Sherr et al. Mitigating dos attack through selective bin verification. NPSEC, 2005.

[62] K. Sriram et al. Enhanced Feasible-Path Unicast Reverse Path Forwarding. RFC 8704, IETF, 2020.

[63] Ion Stoica et al. Core-stateless fair queueing: Achieving approximately fair bandwidth allocations in high speed networks. SIGCOMM, 1998.

[64] Ahren Studer and Adrian Perrig. The coremelt attack. In *Computer Security – ESORICS*, 2009.

[65] Sunrise LLC. SCiON Sunrise Business. http://tinyurl.com/5n6t94tp, 2024.

[66] Swisscom AG. Enhancing WAN connectivity and services for Swiss organisations with the next-generation internet. https://www.swisscom.ch/scion, 2024.

[67] SWITCH. SWITCHlan SCION Access. https://www.switch.ch/scion/, 2024.

[68] Anapaya Systems and ETH Zurich. SCMP specification. https://docs.scion.org/en/latest/protocols/scmp.html, 2024.

[69] Brian Trammell and Dominik Schatzmann. On flow concurrency in the internet and its implications for capacity sharing. CSWS, 2012.

[70] R. ur Rasool et al. A survey of link flooding attacks in software defined network ecosystems. *Journal of Network and Computer Applications*, 2020.

[71] Cun Wang et al. Inferring the average AS path length of the internet. IC-NIDC, 2016.

[72] Bo Wu et al. Enabling efficient source and path verification via probabilistic packet marking. IWQoS, 2018.

[73] Marc Wyss et al. Secure and scalable QoS for critical applications. IWQoS, 2021.

[74] Marc Wyss et al. DoCile: Taming denial-of-capability attacks in inter-domain communications. IWQoS, 2022.

[75] Marc Wyss et al. Protecting critical inter-domain communication through flyover reservations. CCS, 2022.

[76] Z. Yu et al. Twenty years after: Hierarchical core-stateless fair queueing. NSDI, 2021.

[77] M. Zhang et al. Poseidon: Mitigating volumetric DDoS attacks with programmable switches. NDSS, 2020.

A DRKey

DRKey [38, 57] provides symmetric keys between ASes and endpoints and is a requirement both for COLIBRI and EPIC. Instead of simply storing exchanged keys at infrastructure components such as border routers, DRKey enables these components to rapidly compute the keys on the fly. To illustrate this mechanism, we consider some AS A, which maintains a secret key K_A that is known by its infrastructure components such as the control service and all border routers. When another AS B requests a key from AS A, the latter responds with an AS-level symmetric key $\mathrm{K}_{A\to B}$ computed using a pseudorandom function (PRF):

$$\mathrm{K}_{A\to B} := \mathrm{PRF}_{\mathrm{K}_A}(B). \tag{3}$$

After receiving this AS-level key, AS B can use it to derive symmetric keys for its endpoints. It computes symmetric keys between endpoint H_B (in AS B) and AS A as follows:

$$\mathrm{K}_{A\to H_B} := \mathrm{PRF}_{\mathrm{K}_{A\to B}}(H_B), \tag{4}$$

Importantly, endpoint H_B cannot compute $\mathrm{K}_{A\to H_B}$ itself, as it does not have access to $\mathrm{K}_{A\to B}$. Instead, it fetches $\mathrm{K}_{A\to H_B}$ from an infrastructure service in its AS. When traffic originating from endpoint H_B passes through a border router that belongs to AS A, the router can recompute $\mathrm{K}_{A\to H_B}$ based only on its secret key K_A, the AS-identifier B, and the endpoint source address H_B, which are all part of the SCION packet header. By default, keys have a validity of one day.

B EPIC

EPIC [42] utilizes DRKey's efficient key derivation to enable per-packet source authentication in SCION. To authenticate itself to on-path border routers, an endpoint H_S in AS A_0 computes a per-packet hop validation field (HVF) for each AS A_i ($1 \leq \mathrm{i} \leq \mathrm{n}$) on the selected path. These HVFs are included in the SCION packet header.

$$\mathrm{K}_i := \mathrm{K}_{A_i \to H_S} \tag{5}$$

$$\mathrm{V}_i := \mathrm{MAC}_{\mathrm{K}_i}(\mathrm{tsPkt}, A_0, H_S, \sigma_i)\ [0{:}\ell_{\mathrm{val}}] \tag{6}$$

In this context, the function MAC_K generates a message authentication code using the key K, and tsPkt is a unique high-precision timestamp added to the packet header for every packet sent by the source end host. The last input to the MAC, σ_i, is an authenticator included by EPIC to achieve a property called *path authorization*, which protects the routing decision of ASes from malicious end hosts. The notation X[a:b] denotes the substring of X from byte a (inclusive) to byte b (exclusive), and the HVF is defined as the first ℓ_{val} bytes of the MAC output. To verify a packet's source, a border router recomputes the HVF of its AS and compares it to the HVF contained in the packet header. A duplicate-suppression system further allows ASes to filter replayed packets. EPIC also provides assurance to source hosts that their traffic indeed followed the selected path. Path validation is achieved by letting border routers replace the HVFs in the packet with the next ℓ_{val} bytes of the MAC output, i.e., $[\ell_{\mathrm{val}}{:}2\ell_{\mathrm{val}}]$. This serves as proof that the packet indeed traversed the ASes on the selected path. To communicate this information to H_S, the destination endpoint H_D returns a packet that contains the updated HVFs and tsPkt of the original packet.

C Z-Lane Gateway Authorization

Based on DRKey and EPIC, the Z-Lane gateway marks authorized packets such that other ASes' routers can later distinguish them from standard EPIC traffic that did not pass the gateway. This is achieved by replacing the packets HVFs with AS-level authenticated HVFs:

$$\widetilde{\mathrm{K}_i} := \mathrm{K}_{A_i \to A_0} \tag{7}$$

$$\widetilde{\mathrm{V}_i} := \mathrm{MAC}_{\mathrm{K}_i}(\mathrm{V}_i) \tag{8}$$

Here, V_i is the arriving packet's HVF (Eq. 6).[10] Apart from the HVF (and the QoS flag), the gateway does not change any field in the packet header. Overwriting the HVF at the gateway does not break EPIC's path validation feature, because the border router later still updates the HVF to $\mathrm{V}_i[\ell_{\mathrm{val}}{:}2\ell_{\mathrm{val}}]$, i.e., based on the original HVF computed by the end host; thus the destination will still see the original EPIC MACs. Computing $\widetilde{\mathrm{V}_i}$ in addition to V_i increases the overhead at border routers. However, this increase is minimal, as only a single additional AES operation is needed; the key $\widetilde{\mathrm{K}_i}$ is already derived and AES-expanded in the computation of K_i (Eq. 4). We choose this approach of computing $\widetilde{\mathrm{V}_i}$ as a new MAC with input V_i instead of just recomputing V_i according to Equation (6) but with $\mathrm{K}_{A \to B}$, as otherwise an end host would need to also communicate each σ_i to the gateway.

D Memory-Efficient Rate Limiting

In standard literature, the token-bucket rate limiter is either implemented using a counter and a timer or using a counter and a timestamp [49] (plus the configured rate and burst size). The first approach does not scale because for every flow (or every source AS or ISD, in the case of Z-Lane) a separate timer is needed. Also, timers are often not suitable for high-performance applications. The second approach is computationally very efficient but typically not optimized for a low memory footprint: state-of-the-art implementations often require at least 20 B [26]—some even more than 60 B [29]—of memory per token bucket.[11] In this section, we present a data structure that requires only 8 B per token bucket, thus improving Z-Lane's memory-efficiency. Concretely, we do the following optimizations: (i) Configure the same burst size for every token bucket. This means that this value only needs to be stored once. (ii) Store the burst size in 2 B. This allows to define values up to 65 535 B. (iii) Reduce the size of the counter to 2 B. This is possible, because the counter cannot exceed the burst size, on which we put an upper bound in the previous step. (iv) Decrease the size of the rate field to 2 B. With a granularity of 100 kbps, this still allows to configure values up to 6.5 Gbps. (v) Only track short intervals. Instead of storing an 8 B Unix timestamp that measures the time elapsed since 1970, we use 4 B tracking an interval of only 1.3 s. The timestamp still stores time values at nanosecond precision. When monitoring α ASes and υ ISDs, this data structure thus requires only $2\,\mathrm{B} + (\alpha + \upsilon) \cdot 8\,\mathrm{B}$ instead of $(\alpha + \upsilon) \cdot 20\,\mathrm{B}$. In case the configured rates are the same for all monitored ASes and ISDs, they can be replaced by a single globally defined rate, thereby further reducing the per-token-bucket memory overhead by 2 B.

Monitoring Accuracy. Using timestamps to only track 1.3 s intervals can lead to false negatives, where the rate limiter wrongly considers a packet to cause an overuse of the configured rate. However, this situation can only arise if (i) a packet is sent a multiple of 1.3 s after the previous packet and (ii) the previous packet depleted the token bucket. More precisely, this is the case if a packet arrives at a time $\mathrm{t} \in\ ]\lambda \cdot 1.3\,\mathrm{s},\ \lambda \cdot 1.3\,\mathrm{s} + \frac{\ell}{\mathrm{rate}}]$ for $\lambda\ \in \mathbb{N}_{\geq 1}$, where ℓ denotes the packet's length. Therefore, if an AS stops sending any traffic for a random period of at least 1.3 s, then the probability of the next packet being illegitimately dropped is $\frac{\ell}{\mathrm{rate} \cdot 1.3\,\mathrm{s}}$. For

[10] We use an additional, not explicitly mentioned input to the MACs in Equations (4) and (8) to enforce domain separation.

[11] Typical field sizes: 8 B for timestamp, 4 B for counter, burst, and rate.

$\ell \leq 1500\,\mathrm{B}$ and rate $\geq 10\,\mathrm{Mbps}$, this probability is at most 0.093%. If an AS does not stop sending traffic for at least 1.3 s, then packets are never illegitimately dropped.

E Probabilistic Monitoring

Probabilistic monitoring solutions try to reduce the memory overhead inherent to deterministic policing mechanisms (e.g., token buckets). Typically, those solutions first try to detect overuse flows using probabilistic data structures, and then further monitor suspicious flows deterministically, where the number of deterministic monitors is relatively low and upper bounded [59]. Often, overuse flows are then also added to a black list, causing all following traffic belonging to that flow to be dropped. Unfortunately, unlike in bandwidth reservation systems where the traffic sources know the exact traffic rate at which they are allowed to send, traffic sources in Z-Lane do not know the bandwidths configured at on-path ASes, which can lead to unintentional overuse. Also, blacklisting ASes or ISDs that are overusing their configured bandwidth contradicts Z-Lane's goal of achieving communication guarantees. Lastly, if there are more malicious ASes or ISDs than the number of available deterministic monitors, there will be false negatives, meaning that malicious ASes could actually overuse their configured bandwidth and disrupt the supposedly guaranteed communication of other ASes. For the use in Z-Lane, probabilistic systems are therefore not practical.

F Neighbor-Based Scheduling

Scheduling could be neighbor-based instead of source-based. While this would save memory at routers and might not even require source authentication, it is vulnerable to DDoS attacks, where an off-path attacker sends traffic over multiple paths entering different interfaces of the same router, but targeting the same egress. Legitimate traffic then still decreases exponentially with each attacked on-path router.

G Scalability in Large ISDs

To minimize router state in case of exceptionally large ISDs with tens of thousands of ASes, there are multiple options. For topology and routing policy reasons, not every AS can reach every router inside an ISD, therefore guaranteed rates at a router egress interface only need to be configured for ASes that can actually reach that interface. More pragmatically, a router could also simply monitor the ASes responsible for, e.g., 99.9 %, of the usually received traffic. Alternative solutions that do not explicitly lower the communication guarantees of smaller ASes include splitting the ISD into multiple smaller ISDs, or monitoring virtual ISDs, i.e., groups of intra-ISD ASes, e.g., based on AS identifier prefixes.

H Other Source Authentication Systems

Besides EPIC, there are other proposals that aim to achieve source authentication through cryptographic verification. OPT [38] and ICING [50] have significantly higher communication overhead than EPIC, and OPT further only achieves source authentication for routers in a weaker attacker model. PPV [72] only allows the destination host to validate the authenticity of packets, i.e., there is no mechanism in PPV for authenticating the source to on-path routers. Similarly, Hummingbird [33] authenticates a packet's source to the destination host; routers verify packets probabilistically using symmetric keys established with adjacent routers. In MASK [25], only a single on-path router can check the authenticity of the packet's origin. Other work focuses on the problem of authenticated broadcast [31, 61] or multicast [37], and also only authenticates packets to the receiver. Passport [44] leverages routing messages to exchange Diffie-Hellman keys and thus obtain shared secret keys between ASes than are then used for packet source authentication. However, routers in Passport need to cache the AS-level keys and cannot derive them on the fly. Non-cryptographic techniques such as hop count filtering [43] and even current best practices related to source address validation (SAV) [39] such as EFP-uRPF [62] are based on heuristics and therefore cannot provide strict security guarantees.