Tiered Incentives for Integrity Based Queuing

Fariba Khan
University of Illinois at Urbana-Champaign

Carl A. Gunter
University of Illinois at Urbana-Champaign

Abstract

We propose an tiered incentive system called *Integrity-Based Queuing (IBQ)* for protection against Internet Distributed Denial-of-Service (DDoS) attacks. Our proposal can be implemented step-by-step where each integrity improvement brings a direct benefit to the autonomous system making it. IBQ proposes preferential queuing based on integrity: good, bad and middle. Since implementation can rarely be complete or network-wide we provide incremental benefit by prioritizing service for domains with better integrity. We have provided a basic analysis to relate performance to measurable integrity of the client. We have designed the architecture for authentication, queuing and defense. We have tested IBQ for applications with real-time requirements and show how performance improves with higher assurance.

1 Introduction

In the past few years distributed denial-of-service attacks have taken down nations (Georgia, Estonia) [16, 17], disconnected us from social networks (Twitter) [13], and cost businesses billions of dollars [2].

Though the attackers have set up complex underground mechanisms to compromise machines and build botnets, the actual attacks are typically simple. Typically an attacker targets a server and uses its large botnet to send a massive volume of packets with invalid source IP addresses [14]. This makes identification and filtering of attack traffic difficult. Spoofing protections have modest incentives for the party applying them—the main benefits are to the party under attack—and sometimes are too coarse-grained when classifying the origin as valid or invalid.

Ingress filtering [6] at gateways between autonomous systems provides neither complete nor verifiable integrity for the domain. There is nothing to protect the IP of a filtered domain from being spoofed by an attacker in the same domain. More alarming is the fact that there is nothing to protect the IP of a client from a filtered domain from being spoofed by an attacker from any unfiltered domain on the Internet. The victim cannot readily identify an IP address as valid or spoofed. In such a scenario, the source domain has little direct incentive to deploy filtering for its clients.

Advance packet source validation methods such as TVA [23] provide a stronger guarantee of integrity. But they require that many distinct parties, such as competing ISPs, the backbone providers, and servers, collaborate for the best results. Benefits of partial implementation or deployment are not obvious. Parties are verified to be valid or invalid. There is no gradation of validation for parties that have implemented some of the protocol and thus deserve to be in between valid and invalid while getting service. We experimented with a large topology where TVA was implemented by 75% of the nodes. Yet spoofing in the non-participating domains make service unavailable to almost 45% of the legitimate clients and increases service time by five for the rest.

Source address validation and thus DDoS defense could greatly benefit from two measures: (1) incentives for ISPs to deploy integrity mechanisms and (2) gradation of quality for integrity more fine-grained than just proven versus unproven. Non-monetary incentives are well researched in peer-to-peer systems. The incentives ensure that the nodes cooperate at a fair level. We see gradations in software assurance certifications such as EAL levels. Gradation levels make it possible for companies to clearly relate to cost and benefits for upgrading a level or more.

In the Internet it is to be expected that integrity will not be perfect and implementation will be partial. However, even an imperfect implementation can improve the effectiveness of queuing as defense against DDoS attacks when, rather than treating each flow equally, a party with a better integrity level is better treated as an incentive. Our approach is called *Integrity Based Queuing (IBQ)*.

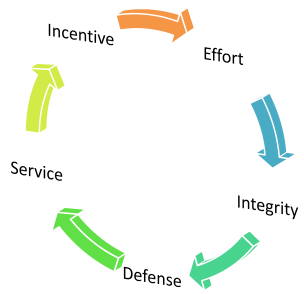


Figure 1: The Cycle of Network Assurance.

IBQ gateways classify packets according to the likelihood that they are from a spoofed origin and allocate bandwidth to high, medium and low integrity flows. Fair queuing for high integrity flows has high effectiveness as each flow gets its own bucket. Gateways impose a differential rate limit while fair queuing medium integrity flows. The rate limit is imposed as a function of the integrity. The low integrity flows receive general queuing.

With IBQ, an ISP that properly authenticates the source IP address of a packet gets the best guarantee of service when the server is under an attack. An ISP that authenticates a packet only to its domain but does not bind the source IP to the packet gets a good guarantee of service but not the best. An ISP that lets its attacking clients spoof the IP addresses on the packets gets service at approximately the rate as it would without IBQ. As ISPs invest more in infrastructure, they may choose to provide better integrity for clients. Better source validation enables them get better service in the face of an attack and works as an incentive for an ISP to spend on integrity-enhancing infrastructure. In Figure 1, we refer to this as *the cycle of integrity assurance*.

This proposal has number of advantages. First, it gives a graded definition of integrity: good, bad and middle. Second, it provides a direct measure of incentive to the ISP: as packets are queued based on integrity, an ISP can see how the performance of applications have improved by increasing a grade of integrity. Third, the middle gradation provides a strategy for ISPs to improve their integrity and performance step-by-step.

2 Related Work

We will discuss about integrity verification, queuing and incentive mechanisms used in DDoS defense.

There are multiple existing solutions for providing integrity but most of these solutions provide a definition of a valid client that is either too narrow or too broad. IPsec [8] provides public-key authentication in IP but the cost of signature verification on core routers and the general complexity of tunnel configuration is hindering to its deployment. In TVA [23] and Pi [21] each router uses a self-verifiable MAC to verify that the initial request packet had passed through it. Passport [10] and StopIt [11] use MAC's with shared keys. These approaches have a free rider problem [9]. Each party needs either some incentive to participate or an enforced payment for rides. For example, if a small fraction of the ISPs invest in an advance protection the general security of the Internet rises, but sufficient benefit is not propagated to the investor for the cost incurred.

The Internet was designed with a fairness criterion. That is every party has an equal right to service and none should starve. Fair-queuing has been an active avenue for congestion control [5, 7, 15, 19, 24]. With the increasing volume of DDoS attacks there has been a renewed interest in fair-queuing algorithms. Approaches such as Pushback [12], TVA [23] use fair-queuing with specially tagged packet flows as part of the protocol. Preferential queuing is used by routers and ISPs to provide QoS required by some service level agreements and for business benefits [20]. But these works do not propose or investigate preferential treatment of packets based on tiered integrity as a protection against DDoS.

In the recent years p2p systems such as file sharing and ad-hoc networks have widely benefited from incentive-based protocols. Internet security, even with all the complex relationships of all the non-cooperating parties, would advance with a good incentive mechanism. Researchers agree that AS-based accountability is key to effective DDoS defense [3, 18].

3 Design

After a packet is sent out by a client from an IBQ domain, the domain gateway router puts an "integrity token" on it identifying the source and destination IP addresses and the originating domain. The integrity token can be verified by the next hop AS in its route. The gateway at the next hop verifies the integrity token and prioritizes the queuing of this packet. The finer origin information the domain provides, the better priority the packet gets. An overview of the architectural support required to provide such abilities are discussed in the next subsections.

Source Integrity Tokens. Traditionally the initial request packets from a client do not bear any proof of origin. IBQ domains attach integrity tokens to their packets. These tokens authenticate the originating domain to the en-route domains. An integrity token, represented as a keyed MAC, is inserted for each AS hop. The key used in the MAC is a shared DH key derived from the public keys of the domains. The public keys can be distributed along BGP announcements [22]. Each token also con-

tains a time stamp and includes part of the packet data so that attackers cannot steal them from valid clients.

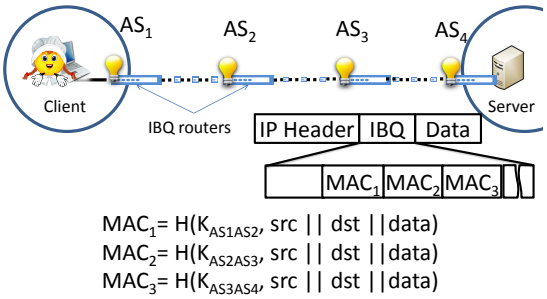


Figure 2: Integrity tokens carried by the packet from the originating AS. It has a MAC for every en-route AS and one for the destination AS.

Figure 2 shows how the integrity tokens are created. If only the client and the server implement integrity tokens IBQ would have little benefit. But as more and more autonomous systems deploy it, the spoofed attacker flow is filtered much closer to the origin.

MACs are used to provide path verification in multiple DDoS defense approaches. Passport [10] and StopIt [11] use a similar mechanism as ours. One could imagine a MAC being added only for the next hop. The next hop after processing the request could add a new MAC verifying itself to be on the route. This adds the burden of signature to the intermediate nodes but has the benefit of path verification for less stable routes. There could be further study in terms of relative cost of both. Yang *et. al.* [22] demonstrates an approach for key management using BGP.

Integrity Levels. The integrity token only authenticates the AS not the source IP address. Attackers could still spoof IP addresses within a domain. On the other hand, though a good 75% of the autonomous systems on the Internet deploy ingress filtering [4], they do not provide any authenticator. To bring the benefits of these two together we propose 'spoofing index table'. A University of Illinois host can spoof 511 neighboring addresses within its $\backslash 23$ prefix. That is it has a 9-bit freedom to spoof an IP address. We define this as *spoofing index* or integrity level for University of Illinois or AS38. The lower this number is the better integrity that AS provides. A *Spoofing Index Table* is a table providing spoofing index information for all autonomous systems.

The MIT ANA Spoofer project [4] tracks the ingress filtering activities of Internet domains. They measure the Internet's susceptibility to spoofed source address IP packets using volunteering clients. Spoofing index information changes rarely and, to get a conservative assessment, the worst integrity level among those reported could be chosen. In essence the data they have collected can act as a spoofing index table. We propose construction of similar software with the data being available publicly, like DNS information, to be used by routers to set the integrity levels for autonomous systems.

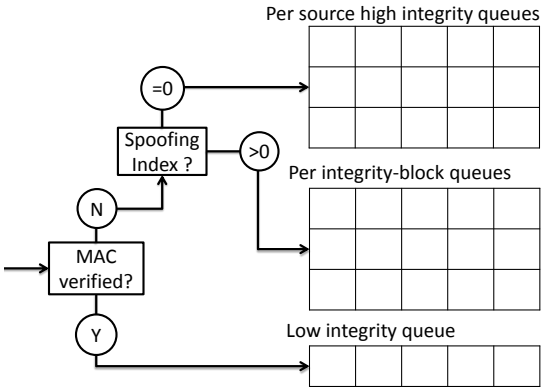
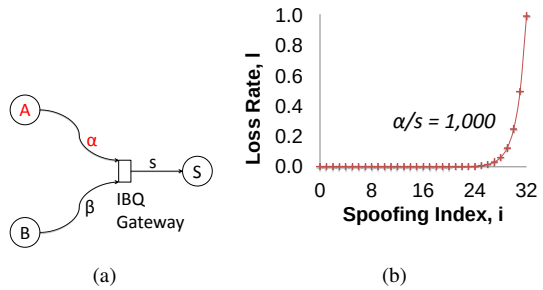


Figure 3: Integrity Based Queuing.

Queuing. The basis of integrity-based queuing is standard fair-queuing where each flow is put into a separate bucket and all the buckets are served in a round robin fashion (assuming equal sized packets). In IBQ, a flow is defined as a group of packets from a common identifiable origin. All packets being identifiable to be from a source are sent into the same bucket. That means when packets arrive from non-IBQ domains and their origin cannot be verified, they receive general queuing. These are the *low integrity* packets.

Packets that come with integrity token for the IBQ router are verified against the spoofing index table for their integrity level. If the home AS has an spoofing index of 0, the flow is classified as having *high integrity* and a filter is created for it. All the high integrity packets are queued in individual per-source buckets. If the spoofing index is higher than zero the packet is in the *middle integrity* category. It is queued with all other packets sharing a IP prefix of ($\backslash 32-$ *spoofing index*) with it. For example, University of Illinois packets from AS38 will be filtered by $\backslash 23$ prefix address. All the packets having the same $\backslash 23$ prefix will be queued together. We call these filters *per-integrity block filters*. Integrity-block filters are part of the fair queue system. But they achieve weighted queuing as the number of sources addresses that are hashed into a filter differs. For example, for an AS that has a spoofing index of 4 only 16 IP addresses hash into a filter whereas for a spoofing index of 8 there are 256 possible sources. But both of these filters are served equally. This means fewer packets from a high spoofing index AS are forwarded.

4 Mathematical Analysis



(a) (b)

Figure 4: Relationship of a AS's spoofing index i and the loss rate experienced by a client for a simple three-node topology.

Consider the simple topology in Figure 4(a). An attacker A and a legitimate client B are communicating with a server S. The legitimate client employs a timeout window and the server can process s requests in a particular time window. We provide an analysis of how a better integrity level enables B to have a better service.

The server is over-provisioned for a legitimate client. Client B makes requests at the rate of β per time window where $\beta \ll s$. That means all requests from the legitimate client are processed by the server when there is no attack. The attacker, on the other hand, seeks to overwhelm the server's capacity to process requests by sending many spurious packets. Say A sends packets at rate α where $\alpha \gg s \gg \beta$. In such a scenario the server can process only portion of the incoming requests. With general queuing the probability of a client request being processed is $\beta/(\alpha+\beta) \approx 0$. The attacker traffic takes over all the capacity of the server. When IBQ is deployed by the server this scenario changes and flows are queued based on their integrity. If all the flows have high integrity the server's capacity is equally shared between them. So A and B both get a fair share of $s/2$. Client B makes many fewer requests than that so all of its requests are processed and it is not overwhelmed by the DDoS attack.

Now let us consider the scenario where B is in a domain with a spoofing index of i. Also consider that the attacker agent is likely to spoof any address on the Internet that it can. The probability that the attacker is spoofing the address of B is one in a few billions (assuming 32-bit address space). But the probability that the attacker is in the same domain as B and will carry the same source authentication is, $p = 1/2^{32-i}$. In that case the probability is, $\beta/(\alpha+\beta)$, which is same as having no defense. So the expected number of packets processed for B is,

$$E(B) = sp\frac{\beta}{\alpha+\beta} + \beta(1-p) = \beta - \beta p + \frac{sp\beta}{\alpha+\beta}$$

So the loss rate l for B is,

$$l = 1 - E(B)/beta = p\left(1 - \frac{s}{\alpha+\beta}\right) \quad (1)$$

If the spoofing index i is close to zero the loss rate is negligible. If there is no source authentication ($i = 32$) there is almost no chance of getting a packet through. But with each grade of integrity (smaller values of i) the chance gets better. This exponential trend is shown in Figure 4(b). This exponential result extends for any topology as DDoS attack rate is always much higher than what the server can process and p dominates.

5 Experimental Evaluation

We use ns2 to analyze IBQ experimentally. We were particularly interested in how real-time applications such as VoIP are affected by IBQ. Traditional circuit-based telephony systems are being replaced by VoIP services on clients (Skype, Vonage) and ISPs (Comcast, Turner). Many ISPs are also teaming up with broadcasters for live-streaming of events (ESPN 360). It is critical for a client that these services perform well under any circumstance. A provider that keeps up the quality even in extreme situations will keep the market share.

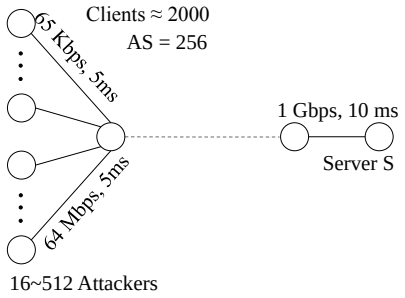


Figure 5: ns2 Topology.

The Internet has grown into billions of nodes and thousands of autonomous systems. Each AS implements a some level of ingress filtering or none. We use the ingress filtering statistics available from the Spoofer [4] project to model our simulations. As ns2 only scales up to few thousand nodes, we scale down the topology accordingly. We simulate 2048 nodes and 256 autonomous systems. We assume only 3% of the nodes from an AS are active clients. The nodes are connected to a high-speed core network with 64Mbps links. The congested link at the server has a capacity of 1Gbps. The link capacities have been scaled down for the scaled down topology.

Attackers use the complete link capacity available to them. They also use their spoofing ability. Attackers are placed uniformly randomly within the clients. For our

topology we scale down the address space to 2^{16} bits. Each AS has a prefix of $\backslash 8$.

We set the parameters of VoIP communication for our clients according to Cisco guidelines [1]. Each client sends packets at a rate of 65kbps for a good quality communication. Each packet is around 190 Bytes. A reasonable quality call should observe less than 1% loss of packets, less than 150ms of delay and a packet delay dispersion of less than 30ms based on International Telecommunication Union-Telecommunication (ITU-T) standards. At this rate the client links and the bottleneck link to the server both are underutilized. Total client bandwidth is about 3Mbps. Without an attack the packets observe an average delay of 50ms and no variation. Loss rate is 0%.

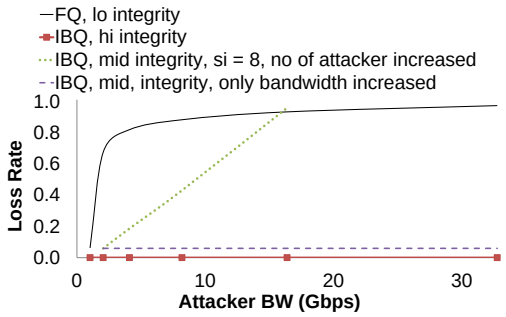


Figure 6: Queuing and source authentication have immense effect on performance.

To compare our approach to base cases, we observe what happens to the VoIP service with changing attack rates. We simulate attacks ranging from 1Gbps to 32 Gbps in bandwidth. The attacks are on a lightly-used link. Though the clients have very good VoIP capability when otherwise that capability vanishes very fast with an attack. Any mechanism that tries to sort out the good and bad packets fail due to the lack of source authentication. We observe that, IBQ performs very well even when using mid-range source validation and only fail when attackers fall in the grey area on the integrity block and choke up the queues. Figure 6 shows the result of this analysis.

Next we compare IBQ to existing methods that deploy source authentication and fairness as a measure of defense. The best comparison should be when both protocols are fully deployed. We experiment with deploying a per-AS fairness in ns2 topology. We use the same parameters as before. The results are shown in Figure 7. Even at its best, a per-AS scheme performs similar IBQ with mid-integrity flows. Attackers distribute their agents globally and a single bot in such a domain could choke all the legitimate clients in that domain.

Partial deployment with per-AS or path queues are

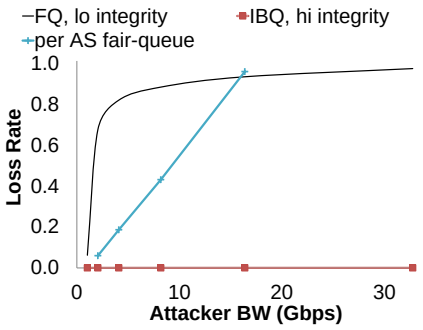


Figure 7: IBQ compared to fully deployed per-AS queuing such as TVA.

tricky. If we imagine the link between the server and the clients to be replaced by a complex topology, how should the quality of authentication mechanism effect the communication. Especially when the client is carrying its own good signature. TVA prefers the newest token on packet rather than the old one while queuing. That might defend against some edge cases of attack, but hampers performance in a regular spoofed packet attack.

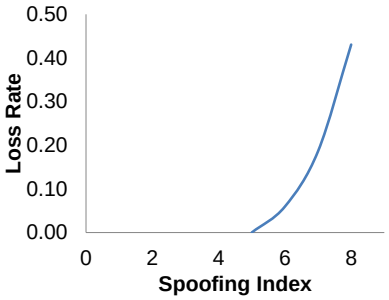


Figure 8: Loss rate as integrity level increases.

In this set of experiments, we fix our attack rate to 8Gbps and relate application performance to the spoofing index. We have shown in § 4 how performance improves exponentially with integrity level. The results in Figure 8 validates that analysis.

One important question that rises from these results is this: should all packets be send through a preferential service or should that service be used only for the request packets? Every protocol starts with few initial packets that request entry into the protocol. Any authentication mechanism used by the application layer starts thereafter. But the tricky part is how to identify and sort those initial packets. In our experiments we observe that the VoIP packets suffer from a loss rate higher than the application can tolerate. So VoIP would benefit from using IBQ to make the call and then establish a secure communication channel. But VoIP, like other network protocols,

comes in many flavors, some open and some proprietary. It would help if request packets were easily recognizable by core-routers.

6 Discussions

Studies [4] show that 25,000 autonomous systems on the Internet and 83.5% of the IP addresses *have an spoofing index of zero*, but hosts and routers generally cannot tell if the packet is from one of these or not. Ingress filtering is not useful unless every netblock uses them. In this paper we give an overview of the design of IBQ to make this coverage effective. Though IBQ has been presented as a defense mechanism for DDoS attacks here, one can readily see how this can be an architecture to provide graded source validation on the Internet. IP spoofing has been a valuable tool for malicious users for a long time.

We have shown analytically and experimentally that IBQ performs well, even with middle-grade integrity. It provides an ISP a direct incentive to deploy this DDoS defense mechanism. Modern edge routers are capable of handling millions of flow filters at line rate. Additionally, IBQ filters or queues are only needed at routers when and where there is congestion. Most protocols will benefit even if IBQ is used only for their initial request packets.

In future we would like to analyze IBQ by itself with different share of integrity levels and see if there is maximum integrity that an ISP would deploy. We want to analyze the stability points of IBQ.

Acknowledgements

This work was supported in part by HHS 90TR0003-01, NSF CNS 09-64392, NSF CNS 09-17218, NSF CNS 07-16626, NSF CNS 07-16421, NSF CNS 05-24695, and grants from the MacArthur Foundation, and Lockheed Martin Corporation. The views expressed are those of the authors only.

References

[1] Voice over ip - per call bandwidth consumption, document id: 7934, Feb 02 2006.

[2] BELSIE, L. Iranian hacker attack: What will it cost twitter? *Christian Science Monitor* (Dec 18 2009).

[3] BENDER, A., SPRING, N., LEVIN, D., AND BHATTACHARJEE, B. Accountability as a service. In *SRUTI'07: Proceedings of the 3rd USENIX workshop on Steps to reducing unwanted traffic on the internet* (Berkeley, CA, USA, 2007), USENIX Association, pp. 1–6.

[4] BEVERLY, R., AND BAUER, S. The spoofer project: inferring the extent of source address filtering on the internet. In *SRUTI'05: Proceedings of the Steps to Reducing Unwanted Traffic on the Internet on Steps to Reducing Unwanted Traffic on the Internet Workshop* (Berkeley, CA, USA, 2005), USENIX Association, pp. 8–8.

[5] DEMERS, A., KESHAV, S., AND SHENKER, S. Analysis and simulation of a fair queueing algorithm. In *SIGCOMM '89: Symposium proceedings on Communications architectures & protocols* (1989), ACM Press, pp. 1–12.

[6] FERGUSON, P., AND SENIE, D. Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing. RFC 2827 (Best Current Practice), May 2000. Updated by RFC 3704.

[7] FLOYD, S., AND FALL, K. Promoting the use of end-to-end congestion control in the internet. *IEEE/ACM Trans. Netw. 7*, 4 (1999), 458–472.

[8] KENT, S., AND SEO, K. Security Architecture for the Internet Protocol. RFC 4301 (Proposed Standard), Dec. 2005.

[9] LELARGE, M., AND BOLOT, J. A local mean field analysis of security investments in networks. In *NetEcon '08: Proceedings of the 3rd international workshop on Economics of networked systems* (New York, NY, USA, 2008), ACM, pp. 25–30.

[10] LIU, X., LI, A., YANG, X., AND WETHERALL, D. Passport: secure and adoptable source authentication. In *NSDI'08: Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation* (Berkeley, CA, USA, 2008), USENIX Association, pp. 365–378.

[11] LIU, X., YANG, X., AND LU, Y. To filter or to authorize: network-layer dos defense against multimillion-node botnets. *SIGCOMM Comput. Commun. Rev. 38*, 4 (2008), 195–206.

[12] MAHAJAN, R., BELLOVIN, S. M., FLOYD, S., IOANNIDIS, J., PAXSON, V., AND SHENKER, S. Controlling high bandwidth aggregates in the network. *SIGCOMM Comput. Commun. Rev. 32*, 3 (2002), 62–73.

[13] MILLS, E. Twitter, Facebook attack targeted one user. *cnet news* (August 6 2009).

[14] MOORE, D., SHANNON, C., BROWN, D. J., VOELKER, G. M., AND SAVAGE, S. Inferring internet denial-of-service activity. *ACM Trans. Comput. Syst. 24*, 2 (2006), 115–139.

[15] NAGLE, J. Congestion control in IP/TCP internetworks. RFC 896 (), Jan. 1984.

[16] RICHARDS, J. Georgia accuses russia of waging cyber-war. *The Times Online* (August 11 2008).

[17] SANGER, D. E., MARKOFF, J., AND SHANKER, T. U.S. Steps up Effort on Digital Defenses. *New York Times Online* (April 28 2009).

[18] SIMON, D. R., AGARWAL, S., AND MALTZ, D. A. AS-based accountability as a cost-effective DDoS defense. In *HotBots'07: Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets* (Berkeley, CA, USA, 2007), USENIX Association, pp. 9–9.

[19] STOICA, I., SHENKER, S., AND ZHANG, H. Core-stateless fair queueing: a scalable architecture to approximate fair bandwidth allocations in high-speed networks. *IEEE/ACM Trans. Netw. 11*, 1 (2003), 33–46.

[20] SVENSSON, P. Comcast blocks some Internet traffic.

[21] YAAR, A., PERRIG, A., AND SONG, D. Pi: A path identification mechanism to defend against DDoS attacks. In *SP '03: Proceedings of the 2003 IEEE Symposium on Security and Privacy* (Washington, DC, USA, 2003), IEEE Computer Society, p. 93.

[22] YANG, X., AND LIU, X. Internet protocol made accountable. In *Eighth ACM Workshop on SIGCOMM Hot Topics in Networks (HotNets-VIII)* (2009), ACM SIGCOMM.

[23] YANG, X., WETHERALL, D., AND ANDERSON, T. TVA: a dos-limiting network architecture. *IEEE/ACM Trans. Netw. 16*, 6 (2008), 1267–1280.

[24] ZHANG, L. Virtual clock: a new traffic control algorithm for packet switching networks. *SIGCOMM Comput. Commun. Rev. 20*, 4 (1990), 19–29.