Proactive Surge Protection: A Defense Mechanism for Bandwidth-Based Attacks

Jerry Chi-Yuan Chou, *Student Member, IEEE*, Bill Lin, *Member, IEEE*, Subhabrata Sen, *Member, IEEE*, and Oliver Spatscheck, *Member, IEEE*

***Abstract*—Large-scale bandwidth-based distributed denial-of-service (DDoS) attacks can quickly knock out substantial parts of a network before reactive defenses can respond. Even traffic that is not under direct attack can suffer significant *collateral damage* if the traffic passes through links that are common to attack routes. This paper presents a *Proactive Surge Protection* (PSP) mechanism that aims to provide a broad first line of defense against DDoS attacks. The approach aims to minimize collateral damage by providing bandwidth isolation between traffic flows. The proposed solution is readily deployable using existing router mechanisms and does not rely on any unauthenticated packet header information. Our extensive evaluation across two large commercial backbone networks, using both distributed and targeted attacks, shows that up to 95.5% of the network could suffer collateral damage, but our solution was able to significantly reduce the amount of collateral damage by up to 97.58% in terms of the number of packets dropped and 90.36% in terms of the number of flows with packet loss. Further, we show that PSP can maintain low packet loss rates even when the intensity of attacks is increased significantly.**

***Index Terms*—Computer networks, network security.**

I. Introduction

A COORDINATED attack can potentially disable a network by flooding it with traffic. Such attacks are also known as bandwidth-based distributed denial-of-service (DDoS) attacks and are the focus of our work. Depending on the operator, the provider network may be a small-to-medium regional network or a large core network. For small-to-medium size regional networks, this type of bandwidth-based attacks has certainly disrupted service in the past. For core networks with huge capacities, one might argue that such an attack risk is remote. However, as reported in the media [9], large botnets already exist in the Internet today. These large botnets combined with the prevalence of high speed Internet access can quite easily give attackers multiple tens of Gb/s of attack capacity. Moreover, core networks are engineered to support normal traffic loads reliably and not to support maximum traffic load from all subscribers. For example, in the Abilene network [1], some of the core routers have an incoming capacity of larger than 30 Gb/s from the access networks, but only 20 Gb/s of outgoing capacity to the core. Although commercial ISPs do not publish their oversubscription levels, they are generally substantially higher than the ones found in the Abilene network due to commercial pressures of maximizing return on investments.

Considering these insights, one might wonder why we have not seen multiple successful bandwidth-based attacks to large core networks in the past. The answer to this question is difficult to assess. Partially, attacks might not be occurring because the organizations which control the botnets are interested in making money by distributing SPAM, committing click frauds, or extorting money from midsized websites. Therefore, they would have no commercial interest in disrupting the Internet as a whole. Another reason might be that network operators are closely monitoring network utilization and actively balancing traffic flow and blocking DDoS attacks. Nonetheless, recent history has shown that if such an attack possibility exists, it will eventually be exploited. For example, SYN flooding attacks were described in [3] years before such attacks were used to disrupt servers in the Internet.

To defend against large bandwidth-based DDoS attacks, a number of defense mechanisms currently exist, but many are reactive in nature (i.e., they can only respond after an attack has been identified in an effort to limit the damage). However, the onset of large-scale bandwidth-based attacks can occur almost instantaneously, causing potentially a huge *surge* in traffic that can effectively knock out substantial parts of a network before reactive defense mechanisms have a chance to respond. To provide a broad first line of defense against DDoS attacks when they happen, we propose a new protection mechanism called Proactive Surge Protection (PSP). In particular, under a flooding attack, traffic loads along attack routes will exceed link capacities, causing packets to be dropped indiscriminately. Without proactive protection, even for traffic flows that are not under direct attack, substantial packet loss will occur if these flows pass through links that are common to attack routes, resulting in significant *collateral damage*. The PSP solution aims to provide *bandwidth isolation* between flows so that the collateral damage to traffic flows not under direct attack is substantially reduced.

This bandwidth isolation is achieved through a combination of traffic data collection, bandwidth allocation of network capacity based on traffic measurements, metering and tagging of packets at the network perimeter into two differentiated priority classes based on capacity allocation, and preferential dropping of packets in the network when link capacities are exceeded. It is important to note that PSP has no impact on the regular operation of the network if no link is overloaded. It therefore introduces no penalty in the common case. In addition, PSP is deployable using existing router mechanisms that are already

Manuscript received June 25, 2008; revised December 06, 2008; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor D. Agrawal. First published August 18, 2009; current version published December 16, 2009. This paper is an extended journal version of a conference paper from the USENIX Security Conference [14] 2008.

J. C.-Y. Chou and B. Lin are with Department of Computer Science, University of California, San Diego, La Jolla, CA 92093 USA (e-mail: jchou@cs.ucsd.edu; billlin@ece.ucsd.edu).

S. Sen and O. Spatscheck are with AT&T Labs-Research, Florham Park, NJ 07974 USA (e-mail: sen@att.research.com; spatsch@att.research.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2009.2017199

available in modern routers, which makes our approach scalable, feasible, and cost effective. Further, PSP is resilient to IP spoofing as well as changes in the underlying traffic characteristics such as the number of TCP connections. This is due to the fact that we focus on protecting traffic between different ingress-egress interface pairs in a provider network and both the ingress and egress interface of an IP datagram can be directly determined by the network operator. Therefore, the network operator does not have to rely on unauthenticated information such as a source or destination IP address to tag a packet.

Specifically, we propose three PSP policies, Mean-PSP, CDF-PSP, and GCDF-PSP. Mean-PSP is solely based on the average traffic demand, while CDF-PSP takes into consideration of the traffic variability observed in historical traffic measurements. CDF-PSP aims to maximize the acceptance probability (or equivalently the min-max minimization of the drop probability) of packets by using the cumulative distribution function over historical data sets as the objective function, and it can be solved as an utility max-min fair bandwidth allocation problem. Finally, GCDF-PSP is a variant of the CDF-PSP policy in which the traffic variability is modeled as a Gaussian distribution, and the problem is simplified to a weighted max-min bandwidth allocation problem. Furthermore, GCDF-PSP allows network operators to model future traffic variability scenarios in which historical datasets are not applicable.

To test the robustness of our proposed approach, we evaluated the PSP mechanism using both *highly distributed* attack scenarios involving a high percentage of ingress and egress routers, as well as *targeted* attack scenarios in which the attacks are concentrated to a small number of egress destinations. Our extensive evaluation across two large commercial backbone networks shows that up to 95.5% of the network could suffer collateral damage, but our solution was able to significantly reduce the amount of collateral damage by up to 97.58% in terms of the number of packets dropped and up to 90.36% in terms of the number of flows with packet loss. In addition, we show that PSP can maintain low packet loss rates even when the intensity of attacks is increased significantly. Beyond evaluating extensively the impact of our protection scheme on packet drops, we also present detailed analysis on the impact of our scheme at the level of flow aggregates between individual ingress-egress interface pairs in the network.

The rest of this paper is organized as follows. Section II outlines related work. Section III presents a high-level overview of our proposed PSP approach. Section IV describes in greater details the central component of our proposed architecture that deals with bandwidth allocation policies. Section V describes our experimental setup, and Section VI presents extensive evaluation of our proposed solutions across two large backbone networks. Finally, we discuss the limitations of the approach in Section VII and conclude the paper in Section VIII.

II. Related Work

DDoS protection has received considerable attention in the literature. The oldest approach, still heavily in use today, is typically based on coarse-grain traffic anomalies detection [2], [25]. Traceback techniques [31], [32], [35] are then used to identify the true attack source, which could be disguised by IP spoofing. After detecting the true source of the DDoS traffic the network operator can block the DDoS traffic on its ingress interfaces by configuring access control lists or by using DDoS scrubbing devices such as [5]. Although these approaches are practical, they do not allow for an instantaneous protection of the network. As implemented today, theses approaches require multiple minutes to detect and mitigate DDoS attacks, which does not match the time sensitivity of today's applications. Similarly, network management mechanisms that generally aim to find alternate routes around congested links also do not match the time sensitivity of today's applications.

Work has also focused on enhancing the current Internet protocol and routing implementations. For example, multiple proposals have suggested to limit the best effort connectivity of the network using techniques such as capabilities models [26], [28], [36], filtering schemes [10], [24], or routing modification [11], [19]. The main focus of these papers is the protection of customers connecting to the core network rather than protecting the core itself, which is the focus of our work. To illustrate the difference, consider a scenario in which an attacker controls a large number of zombies. These zombies could communicate with each other, granting each other capabilities or similar rights to communicate. If planned properly, this traffic is still sufficient to attack a core network. The root of the problem is that the core cannot trust either the sender or the receiver of the traffic to protect itself.

Recently, a couple of novel defense mechanisms deployed in a core network are also proposed to mitigate suspicious attack traffic. One of them is prime [34] and the other is pushback [22]. Similar to the proposals limiting connectivity cited above, prime focuses on protecting individual customers. This leads again to an issue of reliance in that a service provider should not trust its customers for protection. Furthermore, their solution relies heavily on the operator and customers knowing *a priori* who are the good and bad network entities, and their solution has a scalability issue in that it is not scalable to maintain detailed per-customer state for all customers within the network. On the other hand, pushback is a reactive defense mechanism which detects suspicious traffic at congested routes then sends filtering messages to upstream routers. Not only does pushback require communication and cooperation between routers, it simply needs time to react and propagate the filtering messages.

Our work builds on the existing body of literature on max-min fair resource allocation [12], [13], [21], [27], [29], [30], [33], to the problem of proactive DDoS defense. However, our work here is different in that we use max-min fair allocation for the purpose of differential tagging of packets with the objective of minimizing collateral damage when a DDoS attack occurs. Our work here is also different than the server-centric DDoS defense mechanism proposed in [37], which is aimed at protecting end-hosts rather than the network. In their solution, a server explicitly negotiates with selected upstream routers to throttle traffic destined to it. Max-min fairness is applied to set the throttling rates of these selected upstream routers. Like [34] discussed above, their solution also has a scalability issue in that the selected upstream routers must maintain per-customer state for the requested rate limits.

Finally, our work also builds on existing preferential dropping mechanisms that have been developed for providing quality-of-service (QoS) [15], [17]. However, for providing QoS, the service-level-agreements that dictate the bandwidth allocation are

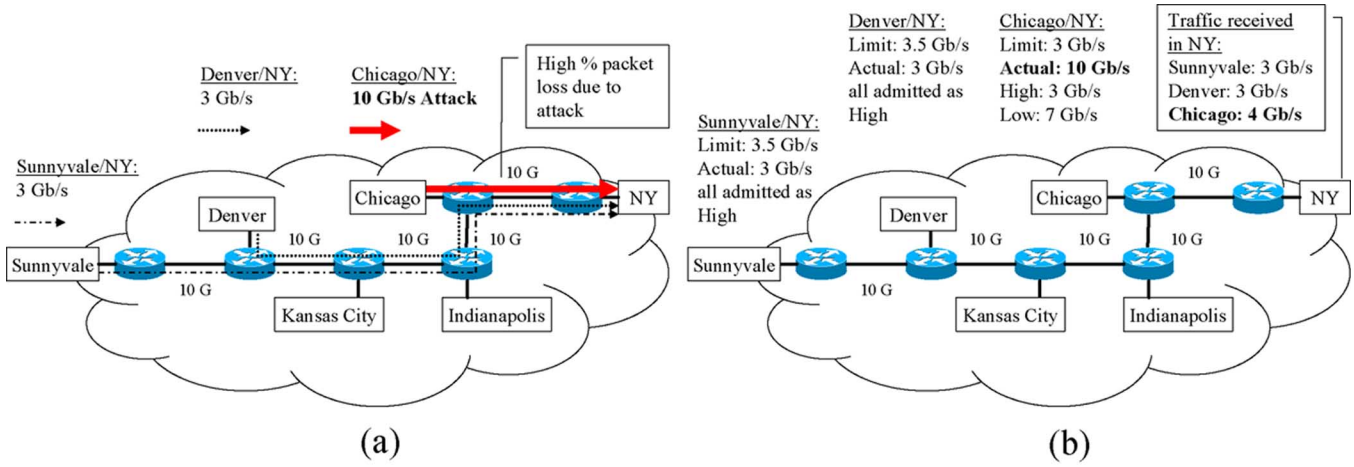


Fig. 1. Attack scenario on the Abilene network. (a) Attack along Chicago/NY. (b) Shielded Sunnyvale/NY and Denver/NY traffic from collateral damage.

assumed to be either specified by customers or decided by the operator for the purpose of traffic engineering. There is also a body of work on measurement-based admission control for determining whether or not to admit new traffic into the network, e.g., [20], [23]. With both service-level-agreement-based and admission-control-based bandwidth reservation schemes, rate limits are enforced. Our work here is different in that we use preferential dropping for a different purpose to provide bandwidth isolation between traffic flows to minimize the damage that attack traffic can cause to regular traffic.

III. Proactive Surge Protection

In this section, we present a high-level architectural overview of a DDoS defense solution called Proactive Surge Protection (PSP). To illustrate the basic concept, we will depict an example scenario for the Abilene network. That network consists of 11 core routers that are interconnected by OC192 (10 Gb/s) links. For the purpose of depiction, we will zoom in on a portion of the Abilene network, as shown in Fig. 1(a). Consider a simple illustrative situation in which there is a sudden bandwidth-based attack along the origin-destination (OD) pair Chicago/NY, where an OD pair is defined to be the corresponding pair of ingress and egress nodes. Suppose that the magnitude of the attack traffic is 10 Gb/s. This attack traffic, when combined with the regular traffic for the OD pairs Sunnyvale/NY and Denver/NY ($3+3+10 = 16$ Gb/s), will significantly oversubscribe the 10 Gb/s Chicago/NY link, resulting in a high percentage of indiscriminate packet drops. Although the OD pairs Sunnyvale/NY and Denver/NY are not under *direct* attack, these flows will also suffer substantial packet loss on links which they share with the attack OD pair, resulting in significant *collateral damage*. The flows between Sunnyvale/NY and Denver/NY are said to be caught in the *crossfire* of the Chicago/NY attack.

A. PSP Approach

The PSP approach is based on providing *bandwidth isolation* between different traffic flows so that the amount of collateral damage sustained along crossfire traffic flows is minimized. This bandwidth isolation is achieved by using a form of *soft* admission control at the perimeter of a provider network. In particular, to avoid saturation of network links, we impose *rate limits* on the amount of traffic that gets injected into the network for each OD pair. However, rather than imposing a *hard* rate limit, where packets are *blocked* from entering the network, we classify packets into two priority classes, *high* and *low*. Metering is performed at the perimeter of the network, and packets are tagged *high* if the arrival rate is below a certain threshold. But when a certain threshold is exceeded, packets will get tagged as *low* priority. Then, when a network link gets saturated, e.g., when an attack occurs, packets tagged with a low priority will be dropped preferentially. This ensures that our solution does not drop traffic unless a network link capacity has indeed been exceeded. Under normal network conditions, in the absence of sustained congestion, packets will get forwarded in the same manner as without our solution.

Consider again the above example, now depicted in Fig. 1(b). Suppose we set the high priority rate limit for the OD pairs Sunnyvale/NY, Denver/NY, and Chicago/NY to 3.5, 3.5, and 3 Gb/s, respectively. This will ensure that the total traffic admitted as high priority on the Chicago/NY link is limited to 10 Gb/s. Operators can also set maximum rate limits to some factor below the link capacity to provide the desired headroom (e.g., set the target link load to be 90%). If the limit set for a particular OD pair is *above* the *actual* amount of traffic along that flow, then all packets for that flow will get tagged as high priority. Consider the OD pair Chicago/NY. Suppose the actual traffic under an attack is 10 Gb/s, which is above the 3 Gb/s limit. Then, only 3 Gb/s of traffic will get tagged as high priority, and 7 Gb/s will get tagged as low priority. Since the total demand on the Chicago link exceeds the 10 Gb/s link capacity, considerable packets would get dropped. However, the packets drop will come from the OD pair Chicago/NY since all packets from Sunnyvale/NY and Denver/NY would have been tagged as high priority. Therefore, the packets for the OD pairs Sunnyvale/NY and Denver/NY would be shielded from collateral damage.

Although our simple illustrative example shown in Fig. 1 only involved one attack flow from one ingress point, the attack traffic in general can be highly distributed. As we shall see in Section VI, the proposed PSP method is also quite effective in such distributed attack scenarios.

B. PSP Architecture

Our proposed PSP architecture is depicted in Fig. 2. The architecture is divided into a policy plane and an enforcement

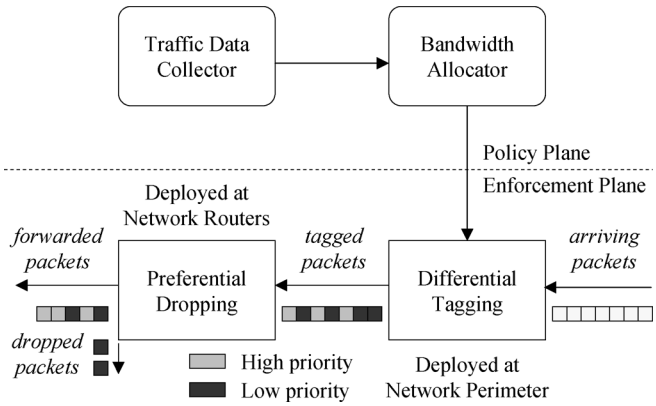


Fig. 2. Proactive surge protection (PSP) architecture.

plane. The traffic data collection and bandwidth allocation components are on the policy plane, and the differential tagging and preferential drop components are on the enforcement plane.

1) Traffic Data Collector: The role of the traffic data collection component is to collect and summarize historical traffic measurements. For example, the widely deployed Cisco sampled NetFlow mechanism can be used in conjunction with measurement methodologies such that those outlined in [18] to collect and derive traffic matrices for different times throughout a day, a week, a month, etc, between different origin-destination (OD) pairs of ingress-egress nodes. The infrastructure for this traffic data collection already exists in most service provider networks. The derived traffic matrices are used to estimate the range of expected traffic demands for different time periods.

2) Bandwidth Allocator: Given the historical traffic data collected, the role of the bandwidth allocator is to determine the *rate limits* at different time periods. For each time period t, the bandwidth allocator will determine a *bandwidth allocation matrix*, $B(t) = [\ b_{s,d}(t)\ ]$, where $b_{s,d}(t)$ is the rate limit for the corresponding OD pair with ingress node s and egress node d for a particular time of day t. For example, a different bandwidth allocation matrix $B(t)$ may be computed for each hour in a day using the historical traffic data collected for same hour of the day. Under normal operating conditions, network links are typically under-utilized. Therefore, traffic demands from historical measurements will reflect this under-utilization. Since there is likely to be *room* for admitting more traffic into the high priority class than observed in the historical measurements, we can fully allocate in some fair manner the available network resources to high priority traffic. By fully allocating the available network resources beyond the previously observed traffic, we can provide *headroom* to account for estimation inaccuracies and traffic burstiness. The bandwidth allocation matrices can be computed offline, and operators can remotely configure routers at the network perimeter with these matrices using existing router configuration mechanisms.

3) Differentiated Tagging: Given the rate limits determined by the bandwidth allocator, the role of the differential tagging component is to perform the metering and tagging of packets in accordance to the determined rate limits. This component is implemented at the perimeter of the network. In particular, packets arriving at ingress node s and destined to egress node d are tagged as high priority if their metered rates are below the threshold given by $b_{s,d}(t)$, using the bandwidth allocation matrix $B(t)$ for the corresponding time of day. Otherwise, they are tagged as low priority. These traffic management mechanisms for metering and tagging are commonly available in modern routers at linespeeds.

4) Preferential Drops: With packets tagged at the perimeter, low priority packets can be dropped preferentially over high priority packets at a network router whenever a sustained congestion occurs. Again, this preferential dropping mechanism [15] is commonly available in modern routers at linespeeds [4], [6]. By using preferential drop at interior routers rather than simply blocking packets at the perimeter when a rate limit has been reached, our solution ensures that no packet gets dropped unless a network link capacity has indeed been exceeded. Under normal network conditions, in the absence of sustained congestion, packets will get forwarded in the same manner as without our surge protection scheme.

IV. Bandwidth Allocation Policies

Intuitively, PSP works by fully allocating the available network resources into the high priority class in some fair manner so that the high priority class rate limits for the different OD pairs are *at least* as high as the *expected* normal traffic. This way, should a DDoS attack occur that would saturate links along the attack route, *normal* traffic corresponding to *crossfire* OD pairs would be *isolated* from the attack traffic, thus minimizing collateral damage. In particular, packets for a particular crossfire OD pair would only be dropped at a congested network link if the *actual* normal traffic for that flow is *above* the bandwidth allocation threshold given to it. Therefore, bandwidth allocation plays a central role in affecting the *drop probability* of normal crossfire traffic during an attack. As such, the goal of bandwidth allocation is to allocate the available network resources with the objective of minimizing the drop probabilities for all OD pairs in some fair manner.

A. Formulation

To achieve the objectives of minimizing drop probability and ensuring fair allocation of network resources, we formulate the bandwidth allocation problem as a utility max-min fair allocation problem [12], [13], [27], [30]. The utility max-min fair allocation problem can be stated as follows. Let $\vec{x} = (x_1, x_2, \ldots, x_N)$ be the allocation to N flows, and let $(\beta_1(x_1), \beta_2(x_2), \ldots, \beta_N(x_N))$ be N utility functions, with each $\beta_i(x_i)$ corresponding to the utility function for flow i. An allocation $\vec{x}$ is said to be *utility max-min fair* if and only if increasing one component x_i must be at the expense of decreasing some other component x_j such that $\beta_j(x_j) \leq \beta_i(x_i)$.

Conventionally, the literature on max-min fair allocation uses the vector notation $\vec{x}(t) = (x_1(t), x_2(t), \ldots, x_N(t))$ to represent the allocation for some time period t. The correspondence to our bandwidth allocation matrix $B(t) = [\ b_{s,d}(t)\ ]$ is straightforward: $b_{s_i,d_i}(t) = x_i(t)$ is the bandwidth allocation at time t for flow i, with the corresponding OD pair of ingress and egress nodes (s_i, d_i). Unless otherwise clarified, we will use the conventional vector notation $\vec{x}(t) = (x_1(t), x_2(t), \ldots, x_N(t))$ and our bandwidth allocation matrix notation interchangeably.

The utility max-min fair allocation problem has been well-studied, and as shown in [13], [30], the problem can be solved by means of a "water-filling" algorithm. We briefly outline here

how the algorithm works. The basic idea is to iteratively calculate the utility max-min fair share for each flow in the network. Initially, all flows are allocated rate $x_i = 0$ and are considered free, meaning that its rate can be further increased. At each iteration, the water-filling algorithm aims to find largest increase in bandwidth allocation to free flows that will result in the maximum common utility with the available link capacities. The provided utility functions, $(\beta_1(x_1), \beta_2(x_2), \ldots, \beta_N(x_N))$, are used to determine this maximum common utility. When a link is saturated, it is removed from further consideration, and the corresponding flows that cross these saturated links are *fixed* from further increase in bandwidth allocation. The algorithm converges after at most L iterations, where L is the number of links in the network, since at least one new link becomes saturated in each iteration. The reader is referred to [13], [30] for detailed discussions.

In the context of PSP, the utility max-min fair algorithm is used to implement different bandwidth allocation policies. In particular, we describe in this section three bandwidth allocation policies, Mean-PSP, CDF-PSP, and GCDF-PSP. All of them are based on traffic data collected from historical traffic measurements. The first policy, Mean-PSP, simply uses the average historical traffic demands observed as *weights* in the corresponding utility functions. Mean-PSP is based on the simple intuition that flows with higher average traffic demands should receive proportionally higher bandwidth allocation. However, this policy does not directly consider the traffic variance observed in the traffic measurements.

To directly account for traffic variance, we propose a second policy, CDF-PSP, that explicitly aims to minimize drop probabilities by using the *cumulative distribution functions* (CDFs) [12] derived from the empirical distribution of traffic demands observed in the traffic measurements. These CDFs can be used to capture the probability that the actual traffic will not exceed a particular bandwidth allocation. When these CDFs are used as utility functions, maximizing the utility corresponds directly to the minimization of drop probabilities.

Finally, GCDF-PSP is proposed when we consider the CDF of historical traffic demands can be approximated by a Gaussian distribution. Specially, we show the utility max-min allocation of CDF-PSP can be reduced to weight max-min in GCDF-PSP by selecting the weight of each flow to be the variance of traffic demand in Gaussian distribution. Each of these three policies is further illustrated next.

B. Mean-PSP: Mean-Based Max-Min Fairness

Our first allocation policy, Mean-PSP, simply uses the mean traffic demand as the utility function. In particular, the utility function for flow i is a simple linear function $\beta_i(x) = \frac{x}{\mu_i}$, where μ_i is the mean traffic demand of flow i, which simplifies to an easier weighted max-min fair allocation problem.

To illustrate how Mean-PSP works, consider the small example shown in Fig. 3. It depicts a simple network topology with 4 nodes that are interconnected by 10 Gb/s links. Consider the corresponding traffic measurements shown in Table I. For simplicity of illustration, each flow is described by just 5 data points, and the corresponding mean traffic demands are also indicated in Table I. Consider the first iteration of the Mean-PSP water-filling procedure shown in Fig. 4(a). The maximum common utility that can be achieved by all *free* flows is $\beta(x) = 1$, which

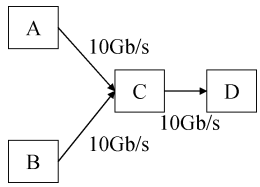


Fig. 3. Network.

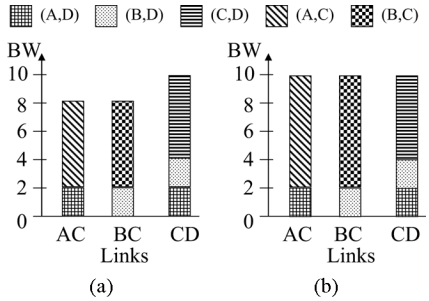


Fig. 4. Mean-PSP water-filling illustrated. (a) 1st iteration. (b) 2nd iteration.

TABLE I
TRAFFIC DEMANDS AND THE CORRESPONDING BANDWIDTH ALLOCATIONS FOR MEAN-PSP AND CDF-PSP

Flows	Historical traffic measurements						BW allocation			
	Measured demands (sorted)					Mean	Mean-PSP		CDF-PSP	
							1st	2nd	1st	2nd
(A,D)	1	1	2	2	4	2	2	2	2	2
(B,D)	1	1	1	3	4	2	2	2	3	3
(C,D)	4	5	5	5	11	6	6	6	5	5
(A,C)	4	5	5	5	11	6	6	8	5	8
(B,C)	5	5	6	6	8	6	6	8	6	7

corresponds to allocating 2 Gb/s each to the OD pairs (A, D) and (B, D) and 6 Gb/s each to the OD pairs (C, D), (A, C), and (B, C). For example, $\beta_{A,D}(x) = \frac{x}{\mu} = 1$ corresponds to allocating $x = 2$ Gb/s since μ for (A, D) is 2. Since all three flows, (A, D), (B, D), and (C, D), share a common link CD, the sum of their first iteration allocation, $2 + 2 + 6 = 10$ Gb/s, would already saturate link CD. This saturated link is removed from consideration in subsequent iterations, and the flows (A, D), (B, D), and (C, D) are fixed at the allocation of 2, 2, and 6 Gb/s, respectively.

On the other hand, link AC is only shared by flows (A, C) and (A, D), which has an aggregate allocation of $2+6 = 8$ Gb/s on link AC after the first iteration. This leaves $10-8 = 2$ Gb/s of *residual* capacity for the next iteration. Similarly, link BC is only shared by flows (B, C) and (B, D), which also has an aggregate allocation of $2 + 6 = 8$ Gb/s on link BC after the first iteration, with 2 Gb/s of residual capacity. After the first iteration, flows (A, C) and (B, C) remain free.

In the second iteration, as in shown Fig. 4(b), the maximum common utility is achieved by allocating the remaining 2 Gb/s on link AC to flow (A, C) and the remaining 2 Gb/s on link BC to flow (B, C), resulting in each flow having 8 Gb/s allocated to it in total. The final Mean-PSP bandwidth allocation is shown in Table I.

C. CDF-PSP: CDF-Based Max-Min Fairness

Our second allocation policy, CDF-PSP, aims to explicitly capture the *traffic variance* observed in historical traffic mea-

surements by using a cumulative distribution function (CDF) model as the utility function. The use of CDFs [12] captures the *acceptance probability* of a particular bandwidth allocation as follows. Let $X_i(t)$ be a random variable that represents the *actual* normal traffic for flow i at time t, and let $x_i(t)$ be the bandwidth allocation. Then the CDF of $X_i(t)$ is denoted as

$$Pr[X_i(t) \leq x_i(t)] = \Phi_{i,t}(x_i(t))$$

and the drop probability is simply the complementary function

$$Pr[X_i(t) > x_i(t)] = 1 - \Phi_{i,t}(x_i(t)).$$

Therefore, when CDFs are used to maximize the acceptance probabilities for all flows in a max-min fair manner, it is equivalent to minimizing the drop probabilities for all flows in a min-max fair manner.

In general, the expected traffic can be modeled using different probability density functions with the corresponding CDFs. One probability density function is to use the empirical distribution that directly corresponds to the historical traffic measurements taken. In particular, let $(r_{i,1}(t), r_{i,2}(t), \ldots, r_{i,M}(t))$ be M measurements taken for flow i at a particular time of day t over some historical data set. Then the empirical CDF is simply defined as

$$\Phi_{i,t}(x_i(t)) = \frac{\#\text{measurements } \leq x_i(t)}{M} = \frac{1}{M}\left(\sum_{k=1}^{M} I(r_{i,k}(t) \leq x_i(t))\right)$$

where $I(r_{i,k}(t) \leq x_i(t))$ is the indicator that the measurement $r_{i,k}(t)$ is less than or equal to $x_i(t)$. For the example shown in Table I, the corresponding empirical CDFs are shown in Fig. 6. For example in Fig. 6(a) for OD pair (A, D), a bandwidth allocation of 2 Gb/s would correspond to an acceptance probability of 80% (with the corresponding drop probability of 20%).

To illustrate how CDF-PSP works, consider again the example shown in Fig. 3 and Table I. Consider the first iteration of the CDF-PSP water-filling procedure shown in Fig. 5(a). To simplify notation, we will simply use for example $\beta_{A,D}(x) = \Phi_{A,D}(x)$ to indicate the utility function for flow (A, D) for some time period t, and we will use analogous notations for the other flows.

In the first iteration, the maximum common utility that can be achieved by all free flows is an acceptance probability of $\beta(x) = 80\%$, which corresponds to allocating 2 Gb/s to (A, D), 3 Gb/s to (B, D), 5 Gb/s each to (C, D) and (A, C), and 6 Gb/s to (B, C). This first iteration allocation is shown in bold black lines in Fig. 6. With this allocation in the first iteration, link CD is again saturated since the sum of the first iteration allocation to flows (A, D), (B, D), and (C, D) is $2+3+5 = 10$ Gb/s, which would already reach the link capacity of CD. Therefore, the saturated link CD is removed from consideration in subsequent iterations, and the flows (A, D), (B, D), and (C, D) are fixed at the allocation of 2, 3, and 5 Gb/s, respectively.

For link AC, which is shared by flows (A, C) and (A, D), the first iteration allocation is $2 + 5 = 7$ Gb/s, leaving $10 - 7 = 3$ Gb/s of residual capacity. Similarly, for link BC, which is shared by flows (B, C) and (B, D), the first iteration allocation is $3 + 6 = 9$ Gb/s, leaving $10 - 9 = 1$ Gb/s of residual capacity.

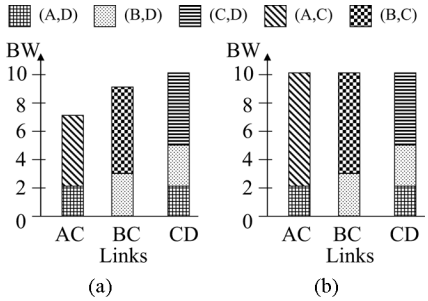


Fig. 5. CDF-PSP water-filling illustrated. (a) 1st iteration. (b) 2nd iteration.

In the second iteration, as in shown Fig. 5(b), the maximum common utility 90% is achieved for the remaining free flows (A, C) and (B, C) by allocating the remaining 3 Gb/s on link AC to flow (A, C) and the remaining 1 Gb/s on link BC to flow (B, C), resulting in a total of 8 Gb/s allocated to (A, C) and 7 Gb/s allocated to (B, C). This second iteration allocation is shown in dotted lines in Fig. 6. The final CDF-PSP bandwidth allocation is shown in Table I.

Comparing the results for CDF-PSP and Mean-PSP shown in Fig. 6 and Table I, we see that CDF-PSP was able to achieve a higher worst-case acceptance probability for all flows than Mean-PSP. In particular, the CDF-PSP results shown in Fig. 6 and Table I show that CDF-PSP was able to achieve a minimum acceptance probability of 80% for all flows whereas Mean-PSP was only able to achieve a lower worst-case acceptance probability of 70%. For example, for flow (B, D), the bandwidth allocation of 3 Gb/s determined by CDF-PSP corresponds to an 80% acceptance rate whereas the 2 Gb/s determined by Mean-PSP only corresponds to a 70% acceptance rate. The better worst-case result is because CDF-PSP specifically targets the max-min optimization of the *acceptance probability* by using the cumulative distribution function as the objective.

D. GCDF-PSP: Gaussian-Based Max-Min Fairness

Another probability Another distribution model that we can use with historical traffic measurements is the Gaussian distribution. GCDF-PSP allows network operators to model future traffic variability scenarios in which historical datasets are not applicable. In addition, we show the problem is simplified to a weighted max-min bandwidth allocation problem.

We denote the CDF for flow i in a Gaussian distribution as $\Phi_{\mu_i,\sigma_i^2}(x)$, where μ_i and σ_i are the mean and standard deviations for flow i, which can be derived from the traffic measurements. To simplify the notation in our bandwidth allocation problem, we will simply use μ_i, σ_i, and $\Phi_{\mu_i,\sigma_i}(x_i)$ to denote the mean, standard deviation, and CDF for a particular time of day t, without any subscript t, unless otherwise needed.

To apply the Gaussian CDF model as utility functions in the water-filling algorithm, at each iteration, instead of finding a bandwidth allocation $\vec{x}(t)_{\bar{p}} = (x_1(t), x_2(t), \ldots, x_N(t))$ that can achieve the largest acceptance probability $\bar{p}$, we find a bandwidth allocation $\vec{x}(t)_{\bar{\lambda}}$ that can achieve the largest scaling factor $\bar{\lambda}$ in the following equation:

$$x_i = \mu_i + \sigma_i \bar{\lambda}.$$

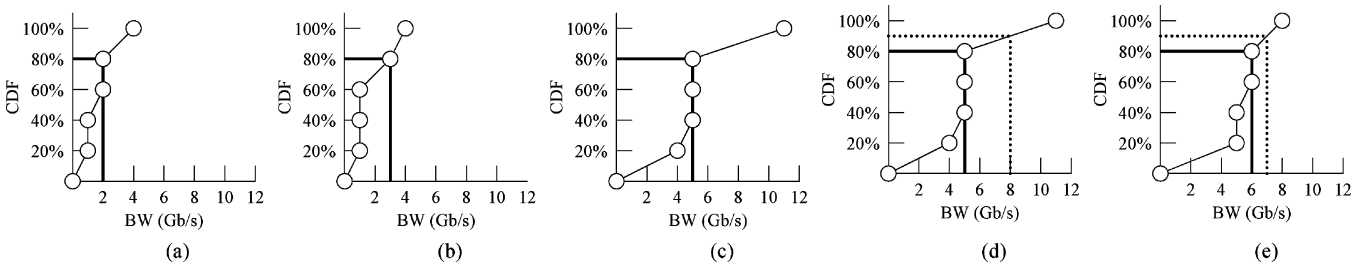


Fig. 6. Empirical CDFs for flows (a) (A, D), (b) (B, D), (c) (C, D), (d) (A, C), and (e) (B, C).

This is equivalent to solving the weighted max-min fair allocation problem using the standard deviations $(\sigma_1, \sigma_2, \ldots, \sigma_N)$ as *weights* and the means $(\mu_1, \mu_2, \ldots, \mu_N)$ as *offsets*.

The reduction rests upon the observation that the inversed CDF for a Gaussian distribution can be written as

$$x_i = \Phi^{-1}_{\mu_i,\sigma_i^2}(p) = \mu_i + \sigma_i\, \Phi^{-1}(p) = \mu_i + \sigma_i\, \sqrt{2}\, \mathrm{erf}^{-1}(2p-1)$$

which is interpreted as the minimum allocation of x_i to ensure an *acceptance probability* of p, where

$$\Phi^{-1}(p) = \sqrt{2}\, \mathrm{erf}^{-1}(2p-1)$$

is the inversed *standard* normal cumulative distribution function defined for $\mu = 0$ and $\sigma = 1$. Therefore, we can see that $\bar{\lambda}$ can be derived from the maximum acceptance probability $\bar{p}$ as follows:

$$\bar{\lambda} = \Phi^{-1}(\bar{p}) = \sqrt{2}\, \mathrm{erf}^{-1}(2\bar{p}-1).$$

Given that $\Phi^{-1}(p)$ is an increasing function, it follows that maximizing $\bar{\lambda}$ in the weighted max-min fair allocation is is equivalent to maximizing $\bar{p}$ in the utility max-min fair allocation problem. This reduction simplifies the bandwidth allocation problem.

V. Experimental Setup

We employed ns-2 based simulations to evaluate our PSP methods on two large real networks.

1) US: This is the backbone of a large service provider in the US, and consists of around 700 routers and thousands of links ranging from T1 to OC768 speeds.

2) EU: This is the backbone of a large service provider in Europe. It has a similar network structure as the US backbone, but it is larger with about 150 more routers and 500 more links.

While the results for the individual networks cannot be directly compared to each other because of differences in their network characteristics and traffic behavior, multiple network environments allow us to explore and understand the performance of our PSP methods for a range of diverse scenarios.

A. Normal Traffic Demand

For each network, using the methods outlined in [18], we build ingress router to egress router traffic matrices from several weeks worth of sampled Netflow data that record the traffic for that network [16]: US (07/01/07–09/03/07) and EU (11/18/06–12/18/06 and 07/01/07–09/03/07). For each time interval τ, the corresponding OD flows are represented by a $N \times N$ traffic matrix where N is the number of access routers providing ingress or egress to the backbone, and each entry contains the average demand between the corresponding routers within that interval. The above traffic data are used both for creating the normal traffic demand for the simulator as well as for computing the corresponding bandwidth allocation matrices for the candidate PSP techniques. One desirable characteristic from a network management, operations and system overhead perspective is to avoid too many unnecessary fine time scale changes. Therefore, one goal of our study was to evaluate the effectiveness of using a single representative bandwidth allocation matrix for an extended period of time. An implicit hypothesis is that the bandwidth allocation matrix does not need to be computed and updated on a fine timescale. To this end, in the simulations, we use a finer timescale traffic matrix with $\tau = 1$ min for determining the normal traffic demand, and a coarser timescale 1-h interval for computing the bandwidth allocation matrix from historical data sets.

B. DDoS Attack Traffic

To test the robustness of our PSP approach, we used two different types of attack scenarios for evaluation – a *distributed* attack scenario for the US backbone and a *targeted* attack scenario for the EU backbone. As we shall see in Section VI, PSP is very effective in both types of attacks. In particular, we used the following.

1) US DDoS: For the US backbone, the attack matrix that we used for evaluation is based on large DDoS alarms that were actually generated by a commercial DDoS detection system deployed at key locations in the network. In particular, among the actual large DDoS alarms there were generated during the period of 6/1/05 to 7/1/06, we selected the largest one involving the most number of attack flows as the attack matrix. This was a *highly distributed* attack involving 40% (nearly half) of the ingress routers as attack sources and 25% of the egress routers as attack destinations. The number of attack flows observed at a single ingress router were up to 150 flows, with an average of about 24 attack flows sourced at each ingress router. The attacks were distributed over a large number of egress routers. Although the actual attacks were large enough to trigger the DDoS alarms, they did not actually cause overloading on any backbone link. Therefore, we scaled up each attack flow to an average of 1% of the ingress router link access capacity. Since there were many flows, this was already sufficient to cause overloading on the network.

2) EU DDoS: For the Europe backbone, we had no commercial DDoS detection logs available. Therefore, we created our own synthetic DDoS attack data. To evaluate PSP under different attack scenarios, we created a *targeted* attack scenario in which all attack flows are targeted to only a small number of egress routers. In particular, to mimic the US DDoS attack

data, we randomly selected 40% of ingress routers to be attack sources. However, to create a targeted attack scenario, we purposely selected at random only 2% of the egress routers as attack destinations. With only 2% of the egress routers involved as attack destinations, we concentrated the attacks from each ingress router to just 1–3 destinations with demand set at 10% of the ingress router link access capacity.

C. Ns-2 Simulation Details

Our experiments are implemented using ns-2 simulations. This involved implementing the 2-class bandwidth allocation, and simulating both the normal and DDoS traffic flows.

1) Bandwidth Allocation and Enforcement: The metering and class differentiation of packets are implemented at the perimeter of each network using the differentiated service module in ns-2, which allows users to set rate limits for each individual OD pair. Our simulation updates the rate limits hourly by precomputing the bandwidth allocation matrix based on the historical traffic matrices that were collected several weeks prior to the attack date: US (07/01/07–09/02/07) and EU (11/18/06–12/17/06 & 07/01/07–09/02/07).

The differentiated service module marks incoming packets into different priorities based on the configured rate limits set by our bandwidth allocation matrix and the estimated incoming traffic rate of the OD pair. Specifically, we implemented differentiated service using TSW2CM (Time Sliding Window with 2 Color Marking), an ns-2 provided policer. As its name implies, the TSW2CM policer uses a sliding time window to estimate the traffic rate.

If the estimated traffic exceeds the given threshold, the incoming packet is marked into the low priority class; otherwise, it is marked into the high priority class. We then use existing preferential dropping mechanisms to ensure that lower priority packets are preferentially dropped over higher priority packets when memory buffers get full. In particular, WRED/RIO[1] is one such preferential dropping mechanism that is widely deployed in existing commercial routers [4], [6], [7], [15]. We used this WRED/RIO mechanism in our ns-2 simulations.

2) Traffic Simulation: For simulation data (testing phase), we purposely used a different data set than the traffic matrices used for bandwidth allocation (learning phase). In particular, for each network, we selected a week-day outside of the days used for bandwidth allocation, and we considered 48 1-min time intervals (one every 30-min) across the entire 24 h of this selected day. The exact date that we selected to simulate normal traffic is 09/03/07 for both the US and EU networks. Recall that for a given time interval τ, we compute normal and DDoS traffic matrices that give average traffic rates across that interval. These matrices are used to generate the traffic flows for that time interval. Both DDoS and network traffic are simulated as constant bandwidth UDP streams with fixed packet sizes of 1 kB.

VI. Experimental Results

We begin our evaluations in Section VI-A by quantifying the potential extent and severity of the problem that we are trying to address—the amount of collateral damage in each network in the absence of any protection mechanism. We then develop an understanding of the damage mitigation capabilities and properties of our PSP mechanism, first at the network level in Section VI-B and then at the individual OD-pair level in Section VI-C. Section VI-D explores the effectiveness of the proposed schemes under scaled attacks, and Section VI-E presents the results under multipath routing.

[1]RIO is WRED with two priority classes.

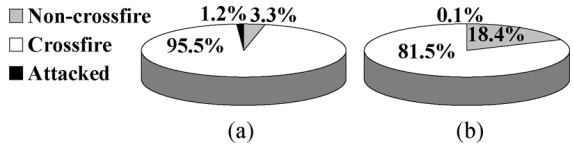


Fig. 7. The percentage of the number of the three OD pair types classified under an attack traffic for (a) US and (b) Europe.

In the following results, we shall use the term No-PSP to refer to the baseline scenario with no surge protection, while we use the terms Mean-PSP, CDF-PSP, and GCDF-PSP to refer to the PSP schemes based on mean, empirical CDF, and Gaussian CDF water-filling bandwidth allocation algorithms respectively. Recall that an OD pair is considered as: 1) an ***attacked OD pair*** if there is attack traffic along that pair; 2) a ***crossfire OD pair*** if it shares at least one link with an OD pair containing attack traffic; and 3) a ***non-crossfire OD pair*** if it is neither an *attacked* nor a *crossfire* OD pair.

A. Potential for Collateral Damage

We first explore the extent to which OD pairs and their offered traffic demands are placed in potential harm's way because they share network path segments with a given set of attack flows. In Fig. 7, we report the relative proportion of OD pairs in the categories of *attacked*, *crossfire*, and *non-crossfire* OD pairs for both the US and EU backbones.

As described in Section V-C, 40% of the ingress routers and 25% of the egress routers were involved in the DDoS attack on the US backbone. In general, for a network with N ingress/egress routers, there are N^2 possible OD pairs (the ratio of routers to OD pairs is 1-to-N). For the US backbone, with about 700 routers, there are nearly half a million OD pairs. Although 40% of the ingress routers and 25% of the egress routers were involved in the attack, the number of attack destinations from each ingress router was on average about 24 egress routers, resulting in just 1.2% of the OD pairs under direct attack. In general, because the number of OD pairs grows quadratically with N (i.e., N^2), even in a highly distributed attack scenario where the attack flows come from all N routers, the number of OD pairs under direct attack may still only correspond to a small percentage of OD pairs. For the EU backbone, there are about 850 routers and about three quarters of million OD pairs. For the targeted attack scenario described in Section V-C, 40% of the ingress routers were also involved in the DDoS attack, but the attacks were concentrated to just 2% of the egress routers. Again, even though 40% of the ingress routers were involved, only 0.1% of the OD pairs, among N^2 OD pairs, were under direct attack.

In general, the percentage of OD pairs that are in the crossfire of attack flows depends on where the attacks occurred and how traffic is routed over a particular network. For the US backbone, we observe that the percentage of crossfire OD pairs is very large (95.5%), causing substantial collateral damage even though the attacks were directed over only 1.2% the OD pairs. This is somewhat expected given the distributed nature of the attack where a high percentage of both ingress and egress routers were involved

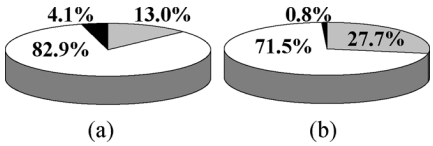


Fig. 8. The proportion of normal traffic demand corresponding to the three types of OD pairs for (a) US and (b) Europe.

in the attack. For the EU backbones, the observed percentage of crossfire OD pairs is also very large (83.5%). This is somewhat surprisingly because the attacks were targeted to only a small number of egress routers. This large footprint can be attributed to the fact that even a relatively small number of attack flows can go over common links that were shared by a vast majority of other OD pairs.

We next depict the relative proportions of the overall normal traffic demand corresponding to each type of OD pairs. While the classification of the OD pairs into the 3 categories is fixed for a given network and attack matrix, the relative traffic demand for the different classes is time-varying, depending on the actual normal traffic demand in a given time interval. Fig. 8 presents a breakdown of the total normal traffic demands for the 3 classes across the 48 time intervals that we explored. Note that for both the networks, crossfire OD pairs account for a significant proportion of the total traffic demand. Figs. 7 and 8 together suggest that an attack directed even over a relatively small number of ingress-egress interface combinations, could be routed around the network in a manner that can impact a significant proportion of OD pairs and overall network traffic.

The results above provide us an indication of the potential "worst-case" impact footprint that an attack can unleash, if its strength is sufficiently scaled up. This is because a crossfire OD pair will suffer collateral packet losses only if some link(s) on its path get congested. While the above results do not provide any measure of actual damage impact, they do nevertheless point to the existence of a real potential for widespread collateral damage, and underline the importance and urgency of developing techniques to mitigate and minimize the extent of such damage.

We next consider the actual collateral damage induced by the specified attacks in the absence of any protection scheme. We define a crossfire OD pair to be *impacted* in a given time interval, if it suffered some packet loss in that interval. Table II presents: 1) the total number of and 2) traffic demand for the impacted OD pairs as a percentage of the corresponding values for all crossfire OD pairs; and 3) the mean packet loss rate across the impacted OD pairs. To account for time variability, we present the average value (with the 10th and 90th percentile indicated in the brackets) for the three metrics across the 48 attacked time intervals. Overall, the tables show that not only can the attacks impact a significant proportion of the crossfire OD pairs and network traffic, but that they can cause severe packet drops in many of them. For example, in the EU network, in 90% of the time intervals: 1) at least 39.64% of the cross-fire OD pairs were impacted; and 2) the average packet loss rate across the impacted OD pairs was 47.62% or more. To put these numbers in proper context, note that TCP, which accounts for the vast majority of traffic today, is known to have severe performance problems once the loss rate exceeds a few single-digit percentage points.

TABLE II
COLLATERAL DAMAGE IN THE ABSENCE OF PSP WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS

	Impacted OD Pairs(%)	Impacted Demand(%)	Mean packet loss rate of impacted OD pairs(%)
US	41.37 [39.64, 42.72]	37.79 [35.16, 39.37]	49.15 [47.62, 50.43]
EU	43.18 [38.48, 47.81]	45.33 [38.90, 52.05]	68.11 [65.51, 70.46]

TABLE III
THE TIME-AVERAGED CROSSFIRE OD-PAIR TOTAL PACKET LOSS RATE WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS

	No-PSP	Mean-PSP	GCDF	CDF-PSP
US	17.93 [16.40, 18.79]	1.63 [1.02, 2.14]	1.49 [0.77, 2.12]	1.11 [0.47, 1.71]
EU	30.48 [27.22, 32.86]	2.73 [1.21, 4.54]	2.72 [1.12, 4.58]	2.32 [0.79, 4.22]

B. Network-Wide PSP Performance Evaluation

We start the evaluation of PSP by focusing on network-wide aggregate performance for crossfire OD pairs and note the consistent substantially lower loss rates under either Mean-PSP, GCDF-PSP or CDF-PSP across the entire day.

1) Total Packet Loss Rate: For each attack time interval, we compute the ***total packet loss rate*** which is the total number of packets lost as a percentage of the total offered load from all crossfire OD pairs. Table III summarizes the mean, 10*th* and 90*th* percentile of the total packet loss rates across 48 attack time intervals. The mean loss rates under No-PSP in US and EU networks are 17.93% and 30.48%, respectively. The loss rate is relatively stable across time as indicated by the tight interval between the 10*th* and 90*th* percentile numbers. In contrast, the mean loss rate is much smaller, less than 3%, for either PSP scheme. Fig. 9 shows the loss rate across time, for the 3 PSP schemes, expressed as a percentage of the corresponding loss rates under No-PSP. Note that even though the attack remains the same over all 48 attack time intervals, the normal traffic demand matrix is time-varying, and hence the observed variability in the time series. In particular, we observe comparatively smaller improvements during the the network traffic peak times, such as 12PM (GMT) in the EU backbone and 6PM (GMT) in the US backbone. This behavior is because the amount of traffic that could be admitted as high priority is bounded by the network's carrying capacity. During high demand time intervals, on one hand, links will be more loaded increasing the likelihood of congestion and overload. On the other hand, more packets will get classified as low priority, increasing the population size that can be dropped under congestion and overload. Table IV summarizes the performance improvements for the PSP schemes in terms of relative loss rate reduction of Mean-PSP and CDF-PSP across the different time intervals. For each network, on average, all PSP schemes reduce the loss rate in a time interval by more than 90% from the corresponding No-PSP value. In addition, CDF-PSP has consistently better performance than Mean-PSP and GCDF-PSP in both networks, while GCDF-PSP has a lower loss rate than Mean-PSP in most of the time. Take US network as an example, CDF-PSP

TABLE IV
THE TIME-AVERAGED TOTAL PACKET LOSS REDUCTION WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS

	Reduction ratio from No-PSP to Mean-PSP	Reduction ratio from No-PSP to CDF-PSP	Reduction ratio from Mean-PSP to CDF-PSP	Reduction ratio from GCDF-PSP to CDF-PSP
US	91.00 [88.56, 93.89]	93.90 [90.77, 97.21]	34.75 [20.06, 53.09]	27.36 [19.47, 38.67]
EU	91.17 [85.79, 96.17]	92.51 [86.46, 97.58]	19.90 [4.01, 41.58]	19.90 [6.21, 35.45]

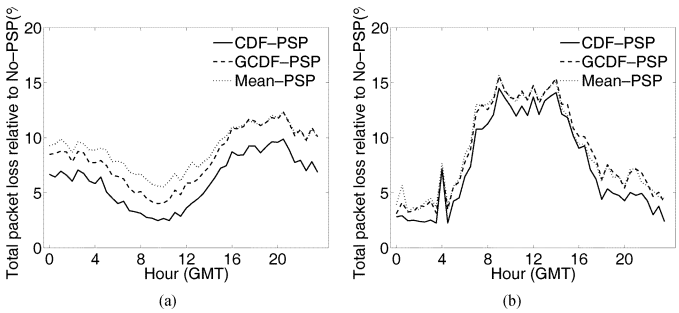


Fig. 9. The crossfire OD pair total packet loss rate ratio over No-PSP across 24 h. (48 attack time intervals, 30 min apart). (a) US. (b) EU.

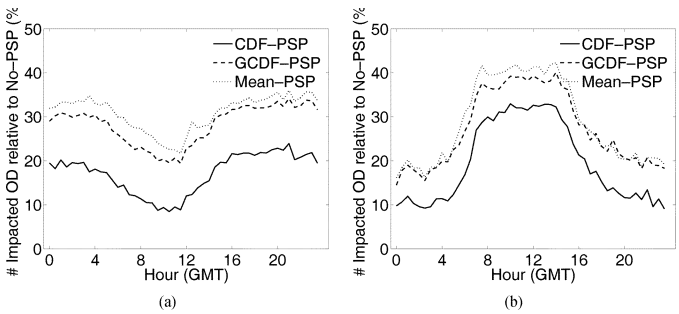


Fig. 10. The ratio of number of crossfire OD-pairs with packet loss over No-PSP across 24 h. (48 attack time intervals, 30 min apart.) (a) US. (b) EU.

reduces the loss rate of Mean-PSP and GCDF-PSP by 34.74% and 27.36, respectively.

2) Number of Impacted Crossfire OD Pairs: We next determine the number of impacted OD pairs, i.e., the crossfire OD pairs that suffer some packet loss at each time interval. It is desirable to minimize this number, since many important network applications including real-time gaming and VOIP are very sensitive to and experience substantial performance degradations even under relatively low packet loss rates. For each of the 48 attack time intervals, we determine the number of impacted crossfire OD pairs as a percentage of the total number of crossfire OD pairs with nonzero traffic demand in that time interval. We summarize the mean and the 10th and 90th percentiles from the distribution of the resulting values across the 48 time intervals in Table V for No-PSP and the three PSP schemes. The mean proportion of impacted OD pairs drops from a high of 41.37% under No-PSP to 12.85% for Mean-PSP, 11.73% for GCDF-PSP, and 7.16% for CDP-PSP. We present the time series of the proportion of impacted OD pairs for the three PSP schemes (normalized by the corresponding value for No-PSP) across the 48 time intervals in Fig. 10, and summarize the savings from the three

TABLE V
THE TIME-AVERAGED NUMBER OF IMPACTED OD-PAIRS WITH PACKET LOSS WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS

	No-PSP	Mean-PSP	GCDF-PSP	CDF-PSP
US	41.37 [39.06, 42.73]	12.85 [9.58, 14.58]	11.73 [8.38, 13.84]	7.16 [3.94, 9.24]
EU	43.18 [38.43, 47.94]	12.81 [7.28, 19.70]	12.10 [7.05, 18.48]	8.79 [3.84, 15.46]

PSP schemes in Table VI. Across all the time intervals, we note that a high percentage of crossfire OD pairs had packet losses under-No-PSP, and that both PSP schemes dramatically reduce this proportion, with CDF-PSP consistently having the lowest proportion of impacted OD pairs. Considering the Table VI, the proportion of impacted OD pairs in the US network is reduced, on average, by over 69% going from No-PSP to Mean-PSP. From Mean-PSP to CDF-PSP, the proportion drops, on average, by a further substantial 45.47%.

C. OD Pair-Level Performance

In Section VI-B, we explored the performance of the PSP techniques from the overall network perspective. We focus the analysis below on the performance of individual crossfire OD pairs across time.

In particular, we analyze the magnitude of packet losses for different crossfire OD pairs. An OD-pair can have different loss rates at different attack time intervals, and here for each crossfire OD pair, we consider the 90th percentile of these loss rates across time, where we consider only time intervals where that OD pair had nonzero traffic demand. Fig. 11 shows the cumulative distribution function (CDF) of this ***90th percentile packet loss rate*** across all crossfire OD-pairs, except those that had no traffic demand during the entire 48 attack time intervals. In the figure, a given point (x, y) indicates that for $y\%$ of crossfire OD-pairs, in 90% of the time intervals in which that OD pair had some traffic demand, the packet loss was at most $x\%$. The most interesting region from a practical performance perspective lies to the left of the graph for low values of the loss rate. This is because many network applications and even reliable transport protocols like TCP have very poor performance and are practically unusable beyond a loss rate of a few percentage points. Focussing on 0%–10% loss rate range which is widely considered to include this 'habitable zone of loss rates', the figure shows that both Mean-PSP, GCDF-PSP and CDF-PSP have substantially higher percentage of OD pairs in this zone, compared to No-PSP, and that CDF-PSP has significantly better performance. For example, the US network, the percentage of OD pair with less than 10% loss rate increases from just 58.84% for No-PSP to 70.48% for Mean-PSP, 74.03% for GCDF-PSP,

TABLE VI
THE TIME-AVERAGED REDUCTION OF NUMBER OF IMPACTED OD-PAIRS HAVING PACKET LOSS WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS

	Reduction ratio from No-PSP to CDF-PSP	Reduction ratio from No-PSP to CDF-PSP	Reduction ratio from Mean-PSP to CDF-PSP	Reduction ratio from GCDF-PSP to CDF-PSP
US	69.05 [65.20, 75.64]	82.82 [78.11, 90.22]	45.47 [35.12, 59.30]	40.30 [31.71, 53.53]
EU	71.18 [58.62, 81.49]	80.42 [67.66, 90.36]	34.94 [21.72, 47.60]	31.45 [16.33, 44.70]

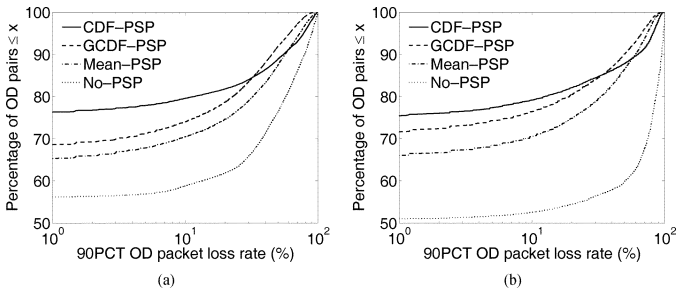


Fig. 11. CDF of the 90 percentile packet loss rate for all crossfire OD pairs. (a) US. (b) EU.

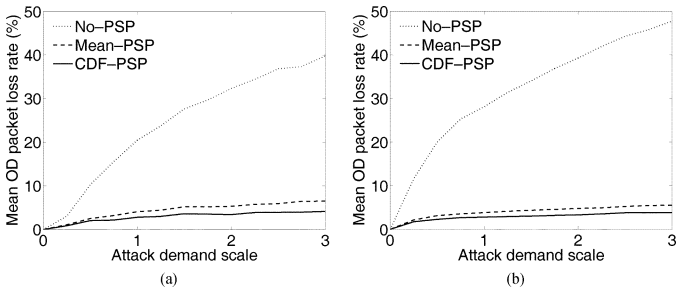


Fig. 12. The time-averaged mean crossfire OD-pair packet loss rate as the attack volume scaling factor increases from 0 to 3. (a) US. (b) EU.

and 79.62% for CDF-PSP. The trends are similar for the EU network.

D. Performance Under Scaled Attacks

Given the growing penetration of broadband connections and the everincreasing availability of large armies of botnets "for hire," it is important to understand the effectiveness of the PSP techniques with respect to increasing attack intensity. To study this, for each network, we vary the intensity of the attack matrix by scaling the demand of every attack flow by a factor ranging from 0 to 3, in steps of size 0.25. For each value of the scaling factor, we measure the time-averaged *mean OD packet loss rate*, which measures the average packet loss rate across all crossfire OD pairs with nonzero traffic demand, across eight 1-min. time intervals, equally spaces across 24 h. Fig. 12 shows that the loss rate under No-PSP increases much faster than under Mean-PSP and CDF-PSP, as the attack intensity increases. This is because under No-PSP, all the normal traffic packets have to compete for limited bandwidth resources with the attack traffic, while with our protection scheme only normal traffic marked in low priority class is affected by the increasing attack. Therefore, even in the extreme case when the attack traffic demand is sufficient to clog all links, our protection scheme can still guarantee that the normal traffic marked in the high priority class

TABLE VII
COLLATERAL DAMAGE UNDER MULTI-PATH IN THE ABSENCE OF PSP WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS. THE DIFFERENCE FROM SINGLE-PATH ROUTING IS INDICATED IN PARENTHESIS

	Impacted OD Pairs(%)	Impacted Demand(%)	Mean packet loss rate of impacted OD pairs(%)
US	37.19 (-4.18) [33.76, 39.10]	36.02 (-1.77) [31.59, 38.32]	47.81 (-1.34) [45.89, 49.73]
EU	44.91(+1.73) [39.50, 48.58]	47.63 (+2.30) [43.20, 50.91]	50.13 (-17.98) [46.10, 54.42]

TABLE VIII
THE TIME-AVERAGED CROSSFIRE OD-PAIR TOTAL PACKET LOSS RATE WITH THE 10TH AND 90TH PERCENTILE INDICATED IN THE BRACKETS. THE DIFFERENCE FROM SINGLE-PATH ROUTING IS INDICATED IN PARENTHESIS

	No-PSP	Mean-PSP	GCDF	CDF-PSP
US	16.33 (-1.60) [14.81, 17.46]	1.46 (-0.17) [0.93, 1.96]	1.34 (-0.15) [0.70, 1.93]	0.98 (-0.13) [0.43, 1.50]
EU	22.60 (-7.88) [19.86, 25.41]	2.28 (-0.45) [1.12, 3.49]	2.10 (-0.62) [0.84, 3.33]	1.93 (-0.39) [0.72, 3.23]

goes through the network. Consequently, our PSP schemes are much less sensitive to the degree of congestion, as evident by the much slower growth of the drop rate. For example, in the US network, as the scale factor increases from 1 to 3, under No-PSP, the mean drop rate jumped from slightly above 20% to almost 40% . In comparison, under CDF-PSP, the mean loss rate increases very little from less than 3% to 4% over the same range of attack intensities. The trends demonstrate that across the range of scaling factor values, both the PSP schemes are very effective in mitigating collateral damage by keeping loss rates low, with CDF-PSP having an edge over Mean-PSP.

E. Multipath Evaluation

Finally, we investigate the impact of multipath routing to our attack scenarios and protection schemes. Specifically, as an example, we consider a Cisco router implementation of a multipath load balancing scheme called Equal-Cost Multipath (ECMP) [8] routing. Here, we revisit the potential of collateral damage in Table VII and the network-wide performance evaluation in Table VIII. Besides the mean, 90th percentile and 10th percentile numbers, we also indicate the difference of mean from the results under single-path experiments.

As shown in Table VII, when a routing algorithm has the ability to route traffic on multiple paths, the degree of damage could be reduced in term of packet loss because link congestion can be alleviated by load balance traffic. For example, the packet loss rate in US and Europe networks were reduced by 1.34% and 17.98%, respectively. However, the range of damage would also be extended because attack traffic is spread across more links. For example, the number of impacted OD pairs and demand

in Europe were increased by 1.73% and 2.30%, respectively. Therefore, while multipath routing could temporarily alleviate the degree of damage, it also creates more potential damage of collateral damage, especially when the volume of attacks could easily be further increased.

We then observe the impact of multipath routing on our schemes. As shown in Table VIII, with multipath routing, the total packet loss rate is reduced under each of our protection schemes. The improvement of our PSP protection scheme in multipath routing is slightly less than single-path routing for two reasons. One reason is multipath routing has greater impact to No-PSP because the packet loss rate of No-PSP is heavily depending on link congestions. Second reason is the traffic load distribution on links in multipath routing may not accurately match to the static load estimation in our bandwidth allocation algorithm. As a result, our PSP scheme could over or under allocate certain link capacity and limit the improvement. Nevertheless, our PSP protection schemes still significantly reduce loss rate and with CDF-PSP performed the best followed by GCDF-PSP and Mean-PSP.

VII. Discussion

Although our PSP protection mechanism could effectively reduce the collateral damage of a DDoS attack, there are some limitations and shortcoming of the approach. In this section, we would like to address each of them as the following. 1) Our approach is designed to reduce the collateral damage of a flooding DDoS attack in a network. While the problem itself is important and interesting to network operators, our approach cannot protect endhost nor defense against those nonbandwidth based attacks which target the network protocol or endhost resources instead of the intermediate network. 2) Our approach is a first-line defense mechanism which aims to effectively mitigate attack damage in a timeliness fashion. While our approach has the strength of scalability and cost, it doesn't have the ability, like previous defense mechanisms, to accurately identify individual attack flows and eliminate them from network by blocking or filtering. Therefore, our approach is orthogonal to those traditional approaches, and we will still recommend to deploy other sophisticated defense systems along with PSP mechanism to further improve network performance. 3) A fundamental limitation of our approach is that we rely on traffic stability to allocate bandwidth without exploring application level packet information. As a result, our approach could treat any bursty traffic as suspicious attack traffic. It will particularly causes problem when the burst traffic is consistent of a bunch of legitimate flows, such as flash crowd. However, the flash crowd traffic would not always be dropped by PSP for two reasons. First, flash crowd traffic only gets dropped when a link congestion actually occurs. Second, because we tend to fully allocate bandwidth based on traffic statistic not simply average traffic, the OD pair with flash crowd traffic could receive higher rate limit. Finally, although flash crowd traffic could be dropped preferentially at congested links, it seems to be a reasonable decision for network operator from the fairness stand point of view because the flash crowd traffic shouldn't grab majority of the bandwidth regardless the users are legitimate or not.

VIII. Conclusion

PSP provides network operators with a broad first line of proactive defense against DDoS attacks, significantly reducing the impact of sudden bandwidth-based attacks on a service provider network. Among its salient features, PSP is readily deployable using existing router mechanisms, and PSP does not rely on any unauthenticated packet header information. The latter feature makes the solution resilient to evading attack schemes that launch many seemingly legitimate TCP connections with spoofed IP addresses and port numbers. By taking into consideration traffic variability observed in traffic measurements, our proactive protection solution can ensure the maximization of the acceptance probability of each flow in a max-min fair manner, or equivalently the minimization of the drop probability in a min-max fair manner. Our extensive evaluation across two large commercial backbone networks, using both distributed and targeted attacks, shows that up to 95.5% of the network could suffer collateral damage even though the attacks were directed over only 1.2% of the OD pairs. Our solution was able to significantly reduce the amount of collateral damage by up to 97.58% in terms of the number of packets dropped and 90.36% in terms of the number of flows with packet loss. In addition, we show that PSP can maintain low packet loss rates even when the intensity of attacks is increased significantly, and PSP has similar protection performance under multipath routing.

References

[1] *Advanced Networking for Leading-Edge Research and Education*, [Online]. Available: http://abilene.internet2.edu

[2] *Arbor Peakflow*, [Online]. Available: www.arbor.net

[3] CERT CA-1996–21 TCP SYN Flooding and IP Spoofing Attacks.

[4] *Cisco CRS-1*, [Online]. Available: http://www.cisco.com/en/US/products/ps5763/index.html

[5] *Cisco Guard*, [Online]. Available: http://www.cisco.com/en/US/products/ps5888/index.html

[6] *Cisco IOS XR Software Release 3.6.0 for Cisco crs-1 Routers*, [Online]. Available: http://www.cisco.com/en/US/products/ps5763/index.html

[7] *Distributed Weighted Random Early Detection*, [Online]. Available: http://www.cisco.com/univercd/cc/td/doc/product/software/ios111/cc111/wred.pdf

[8] "Analysis of an equal-cost multi-path algorithm," RFC2992.

[9] "The Botnet trackers," *Washington Post*, 2006.

[10] K. Argyraki and D. R. Cheriton, "Active Internet traffic filtering: Real-time response to denial-of-service attacks," in *Proc. Annu. Conf. USENIX Ann. Tech. Conf.*, Berkeley, CA, 2005, pp. 10–10.

[11] H. Ballani, Y. Chawathe, S. Ratnasamy, T. Roscoe, and S. Shenker, "Off by default!," presented at the ACM HotNets Workshop, Nov. 2005.

[12] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[13] Z. Cao and E. W. Zegura, "Utility max-min: An application-oriented bandwidth allocation scheme," in *Proc. IEEE INFOCOM*, 1999, pp. 793–801.

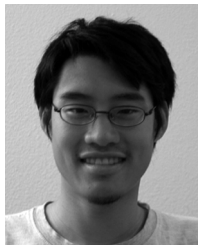
[14] J. Chou, B. Lin, S. Sen, and O. Spatscheck, "Proactive surge protection: A defense mechanism for bandwidth-based attacks," in *Proc. 17th USENIX Security Symp.*, Jul. 2008, pp. 123–138.

[15] D. Clark and W. Fang, "Explicit allocation of best-effort packet delivery service," *IEEE/ACM Trans. Netw.*, vol. 6, no. 4, pp. 362–373, Aug. 1998.

[16] N. G. Duffield, C. Lund, and M. Thorup, "Estimating flow distributions from sampled flow statistics," in *Proc. ACM SIGCOMM*, Aug. 2003, pp. 325–336.

[17] M. A. El-Gendy, A. Bose, and K. G. Shin, "Evolution of the Internet QoS and support for soft real-time applications," *Proc. IEEE*, vol. 91, pp. 1086–1104, Jul. 2003.

[18] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True, "Deriving traffic demands for operational IP networks: Methodology and experience," in *Proc. ACM SIGCOMM*, Jun. 2000, pp. 257–270.
[19] A. Greenhalgh, M. Handley, and F. Huici, "Using routing and tunneling to combat dos attacks," in *Proc. SRUTI Steps to Reducing Unwanted Traffic on the Internet Workshop*, Berkeley, CA, 2005, pp. 1–1.
[20] M. Grossglauser and D. N. C. Tse, "A framework for robust measurement-based admission control," *IEEE/ACM Trans. Netw.*, vol. 7, no. 3, pp. 293–309, Jun. 1999.
[21] Y. Hou, H. Tzeng, and S. Panwar, "A generalized max-min rate allocation policy and its distributed implementationusing the ABR flow control mechanism," in *Proc. IEEE INFOCOM*, 1998, pp. 1366–1375.
[22] J. Ioannidis and S. M. Bellovin, "Implementing pushback: Router-based defense against DDoS attacks," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, Reston, VA, Feb. 2002, pp. 100–108.
[23] S. Jamin, P. B. Danzig, S. Shenker, and L. Zhang, "A measurement-based admission control algorithm for integrated services packet networks," *IEEE/ACM Trans. Netw.*, vol. 5, no. 1, pp. 56–70, Feb. 1996.
[24] K. Lakshminarayanan, D. Adkins, A. Perrig, and I. Stoica, "Taming IP packet flooding attacks," presented at the ACM HotNets Workshop, 2003.
[25] X. Li, F. Bian, M. Crovella, C. Diot, R. Govindan, G. Iannaccone, and A. Lakhina, "Detection and identification of network anomalies using sketch subspaces," in *Proc. ACM/USENIX IMC*, Oct. 2006, pp. 147–152.
[26] B. Parno, D. Wendlandt, E. Shi, A. Perrig, B. Maggs, and Y.-C. Hu, "Portcullis: Protecting connection setup from denial-of-capability attacks," *Comput. Commun. Rev.*, vol. 37, no. 4, pp. 289–300, 2007.
[27] B. Radunovic and J.-Y. L. Boudec, "A unified framework for max-min and min-max fairness with applications," *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 1073–1083, Oct. 2007.
[28] B. Raghavan and A. C. Snoeren, "A system for authenticated policy-compliant routing," in *Proc. ACM SIGCOMM*, Oct. 2004, pp. 167–178.
[29] J. Ros and W. Tsai, "A theory of convergence order of max-min rate allocation and an optimal protocol," in *Proc. IEEE INFOCOM*, 2001, pp. 717–726.
[30] D. Rubenstein, J. Kurose, and D. Towsley, "The impact of multicast layering on network fairness," *IEEE/ACM Trans. Netw.*, vol. 10, no. 2, pp. 169–182, Apr. 2002.
[31] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Network support for IP traceback," *IEEE/ACM Trans. Netw.*, vol. 9, no. 3, pp. 226–237, Jun. 2001.
[32] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer, "Single-packet IP traceback," *IEEE/ACM Trans. Netw.*, vol. 10, no. 6, pp. 721–734, Dec. 2002.
[33] H. Tzeng and K. Siu, "On max-min fair congestion control for multicast ABR service in ATM," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 3, pp. 545–556, 1997.
[34] P. Verkaik, O. Spatscheck, J. V. der Merwe, and A. C. Snoeren, "Primed: Community-of-interest-based ddos mitigation," in *Proc. ACM LSAD Workshop*, Nov. 2006, pp. 147–154.
[35] A. Yaar, A. Perrig, and D. Song, "Pi: A path identification mechanism to defend against DDoS attacks," in *Proc. IEEE Security Privacy Symp.*, May 2003, pp. 93–107.
[36] A. Yaar, A. Perrig, and D. Song, "An endhost capability mechanism to mitigate DDoS flooding attacks," in *Proc. IEEE Security Privacy Symp.*, May 2004, pp. 130–143.
[37] D. K. Y. Yau, J. C. S. Lui, F. Liang, and Y. Yam, "Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 29–42, Feb. 2005.



Jerry Chi-Yuan Chou (S'03) received the B.S. and M.S. degrees in computer science from Tsing Hua University, Hsinchu, Taiwan.

Currently, he is pursuing the Ph.D. degree under advisor Bill Lin with the Department of Computer Science and Engineering, University of California, San Diego. His research area is system and networking, particularly in issues on resource allocation, network routing and system performance, etc.

Bill Lin (M'97) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley.

He is currently on the faculty of Electrical and Computer Engineering, University of California, San Diego, where he is actively involved with the Center for Wireless Communications (CWC), the Center for Networked Systems (CNS), and the California Institute for Telecommunications and Information Technology (CAL-IT^2) in industry-sponsored research efforts. His research has led to more than 100 journal and conference publications. He also holds two patents.

Subhabrata Sen (M'08) received the B.E. degree in computer science from Jadavpur University, Kolkata, India, and the M.S. and Ph.D. degrees in computer science from the University of Massachusetts at Amherst.

He is a Member of the Networking Research Department, Internet and Networking Systems Research Center, AT&T Labs Research, Florham Park, NJ. His research interests lie in Internet technologies and applications and span IP network management, traffic measurement characterization and classification, network data mining, security and anomaly detection, peer-peer systems, and end-to-end support for streaming multimedia.

Oliver Spatscheck (M'03) received the Ph.D. degree in computer science from the University of Arizona, Tucson, in 1999.

He is a Senior Technical Specialist with AT&T Labs-Research, Florham Park, NJ, where he has been actively involved in network monitoring, DDOS analysis, and content distribution. His research publications span the areas of security, network measurement, operating systems, and CDNs—including a book on Web caching and replication. He has also been active in the IETF, working most recently on content distribution networking coauthoring RFC3568. Currently, he focuses his activities on making networks more survivable in today's Internet.