Routing Bottlenecks in the Internet – Causes, Exploits, and Countermeasures

Min Suk Kang and Virgil D. Gligor

May 15, 2014

CMU-CyLab-14-010

CyLab
Carnegie Mellon University
Pittsburgh, PA 15213

Routing Bottlenecks in the Internet – Causes, Exploits, and Countermeasures

Min Suk Kang
ECE and CyLab
Carnegie Mellon University
Pittsburgh, PA USA
minsukkang@cmu.edu

Virgil D. Gligor
ECE and CyLab
Carnegie Mellon University
Pittsburgh, PA USA
gligor@cmu.edu

ABSTRACT

How pervasive is the vulnerability to link-flooding attacks that degrade connectivity of thousands of Internet hosts? Are some network topologies and geographic regions more vulnerable than others? Do practical countermeasures exist? To answer these questions, we introduce the notion of the *routing bottlenecks* and show that it is a fundamental property of Internet design; i.e., it is a consequence of route-cost minimizations. We illustrate the pervasiveness of routing bottlenecks in an experiment comprising 15 countries and 15 cities distributed around the world, and measure their susceptibility to link-flooding attacks. We present the key characteristics of routing bottlenecks, including size, link type, and distance from host destinations, and suggest specific structural and operational countermeasures to link-flooding attacks. These countermeasures can be deployed by network operators without major Internet redesign.

1. INTRODUCTION

Recent experiments [24] and real-life attacks [7] have offered concrete evidence that link-flooding attacks can severely degrade, and even cut off, connectivity of large sets of adversary selected hosts in the Internet for uncomfortably long periods of time; e.g., hours. However, neither the root cause nor pervasiveness of this vulnerablity has been analyzed to date. Furthermore, it is unknown whether certain network structures and geographic regions are more vulnerable to these attacks than others. In this paper we address this gap in our knowledge about these attacks by (1) defining the notion of the *routing bottlenecks* and its role in enabling link-flooding attacks at scale; (2) finding bottlenecks in 15 countries and 15 cities distributed around the world to illustrate their pervasiveness; and (3) measuring bottleneck parameters (e.g., size, link types, and distance to adversary-selected hosts) to understand the magnitude of attack vulnerability. We also present both structural and operational countermeasures that mitigate link-flooding attacks.

In principle, route diversity could enhance Internet resilience to link-flooding attacks against *large sets* of hosts (e.g., 1,000 hosts) since it could force an adversary to scale attack traffic to unattainable levels to flood all possible routes. In practice, however, the mere existence of many routes between traffic sources and selected sets of destination hosts cannot guarantee resilience whenever the vast majority of these routes are distributed across very few links, which could effectively become a routing bottleneck. To define routing bottlenecks more precisely, let S denote a set of (source) IP addresses of hosts that originate traffic to a set of IP destination addresses, denoted by D. S represents any set of hosts distributed across the Internet. In contrast, D represents a set of hosts of a specified Internet region (e.g., a country or a city), which are chosen at random and independently of S. A *routing bottleneck* on the routes from S to D is a small set B of IP (layer-3) links such that B's links are found in a majority of routes whereas the remaining links are found in very few routes. $|B|$ is often over an order of magnitude smaller than both $|S|$ and $|D|$. If all links are ranked by their frequency of occurrence in the routes between S and D, the bottleneck links, B, have a very high rank whereas the vast majority of the remaining links have very low rank. The sharper the skew in the frequency of link occurrence in these routes, the narrower the bottleneck. Routes may have more than one bottleneck of size $|B|$.

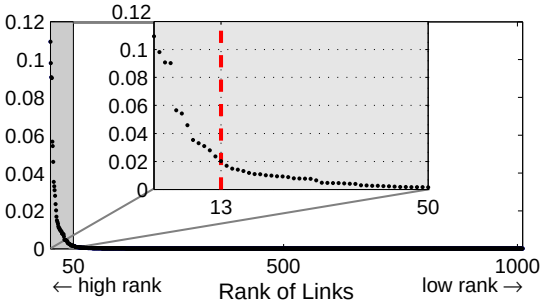

Figure 1: Normalized link-occurrence distribution in routes from S = 250 PlanetLab nodes to D = 1,000 randomly selected servers in *Country15*.

An Example. To illustrate a real routing bottleneck, we represent route sources S by 250 PlanetLab nodes [39] distributed across 164 cities in 39 countries. For the route destinations, D, we select 1,000 web servers at random from a list of publicly-accessible servers obtained using the 'computer search engine' called Shodan (`http://www.shodanhq.com`) in each of the selected Internet region; i.e., in *Country15* of the fifteen-country list $\{Country1, \cdots, Country15\}$. This list is a permutation of the alphabetically ordered list of countries {Brazil, Egypt, France, Germany, India, Iran, Israel, Italy, Japan, Romania, Russia, South Korea, Taiwan, Turkey, and United Kingdom}.[1] We trace the routes between S and D for *Country15*, collect over 1.9×10^6 link samples from those routes, and plot their link-occurrence distribution,[2] as

[1]The permutation is a country ordering by link-occurrence skew. Finding it and de-anonymizing the country list would require repeating the measurements illustrated in Fig. 2.
[2]When analyzing the occurrence of the links, we remove

shown in Fig. 1. This figure clearly shows a very skewed link-occurrence distribution, which implies the existence of a narrow routing bottleneck; i.e., $|B| = 13$ links are found in over 72% of the routes; viz., Fig. 11.

In this paper we argue that the pervasive occurrence of routing bottlenecks is a fundamental property of the Internet design. That is, power-law distributions that characterize the frequency of link occurrence in routes are a consequence of employing route-cost minimization, which is a *very desirable* feature of Internet routing[3]; viz., discussion of BGP and OSPF routing in Sections 2 and 5. Fortunately, routing bottlenecks do not lead to traffic degradation during ordinary Internet use, because the capacity of bottleneck links is usually provisioned adequately for normal mode of operation.

Problem. Unfortunately, however, bottleneck links provide a very attractive target to an adversary whose goal is to flood few links and severely degrade or cut off connectivity of targeted servers, D, in various cities or countries around the world. For example, an adversary could easily launch a traffic amplification attack using NTP monlists (400 Gbps) [35] and DNS recursors (120 Gbps) [7] to distribute an aggregate of 520 Gbps traffic across the 13-link bottleneck of Fig. 1. Such an attack would easily flood these links, even if each of them is provisioned with a maximum of 40 Gbps capacity, severely degrading the connectivity of the 1,000 servers of *Country15* from the Internet; viz., Fig. 11. More insidious attacks, such as Crossfire [24], can flood bottleneck links with traffic that is indistinguishable from legitimate; viz., Section 6 - Related Work.

To counter link-flooding attacks that exploit routing bottlenecks, we first define the parameters that characterize these bottlenecks; e.g., size, link types, and average distance of bottleneck links from the targeted servers, D. Then we define a *connectivity-degradation metric* to provide a quantitative view of the risk exposure faced by these servers. The bottleneck parameters and metric are particularly important for applications in the targeted country or city where Internet-facing servers need stable connectivity; e.g., industrial control systems [9], financial [45], defense and other government services. For these applications, routing bottlenecks pose unexpected vulnerabilities, since diversity of IP-layer connections, which is often incorrectly believed to be sufficient for route diversity, only assures necessary conditions for route diversity but does not guarantee it. We illustrate the usefulness of our connectivity-degradation metric in assessing the vulnerabilities posed by real life routing bottlenecks found in fifteen countries and fifteen different cities around the world; viz., Section 2 and Appendix A.

Analysis of routing bottleneck exploits explains why intuitive but naive countermeasures will not work in practice; e.g., reactive re-routing to disperse the traffic flooding bottleneck links across multiple local links; flow filtering at routers based on traffic intensity; reliance on backup links on exposed routes. More importantly, our analysis provides a precise *route-diversity metric*, which is based on autonomous-system (AS) path diversity, and illustrates the utility of this metric as a proxy for the bottleneck avoidance in the Internet. Finally, our analysis suggests operational countermeasures against link-flooding attacks, including inter- and intra-domain load balancing, and automatic intra-domain traffic engineering.

Contributions. In summary, we make the following contributions:

- We explain the root causes and the characteristics of routing bottlenecks in the Internet, and illustrate their pervasiveness with examples found in 15 countries and 15 cities around the world.
- We present a precise quantitative measure of connectivity degradation to illustrate how routing bottlenecks enable an adversary to scale link-flooding attacks without much additional attack traffic.
- We present several classes of countermeasures against attacks that exploit routing bottlenecks, including both structural and operational countermeasures.

2. ROUTING BOTTLENECKS

2.1 Link-occurrence measurements

To determine the existence of routing bottlenecks, we measure the link-occurrence distribution in a large number of the routes towards a selected destination region. This requires that we perform *traceroute* to obtain a series of *link samples* (i.e., IP addresses of layer-3 links) on a particular route from a source host to a destination host in a selected Internet region. From the collected link samples, we construct the link occurrence distribution by counting the number of samples for each link. In the calculation of link occurrence, we remove the redundant links, which share majority of serving routes in common, from the dataset (viz., Section 4.1 for specific algorithms). In these measurements, for each country or city, we traced 250,000 routes by using *traceroute* from 250 source hosts (i.e., 250 PlanetLab nodes [39]) to 1,000 randomly selected web servers in each of 15 countries.

Traceroute is a commonly used but frequently misused network monitoring tool [43]. We avoid common potential pitfalls (e.g., alias resolution, load-balanced routes, accuracy of returned IP, hidden links in MPLS tunnels and etc.) when we analyze the output of *traceroute*. For detailed discussion, viz., Section 3.3. We perform multiple *traceroutes* for the same source-destination host pair to determine the *persistent* links; i.e., links that always show up in the multiple *traceroutes*. We collect only the samples of persistent links because non-persistent links do not lead to reliable exploitation of routing bottleneck. We have found extremely skewed link-occurrence distribution for the 1,000 randomly selected hosts in each of the 15 countries and this strongly indicates the existence of routing bottlenecks in all the countries in which we performed our measurements. We found similarly skewed distributions when we chose 1,000 random hosts in each of the 15 geographically-distributed cities around the world; viz., Appendix A.

2.2 Power-law in link occurrence distributions

The analysis of link-occurrence distributions helps us understand both the cause of routing bottlenecks and their physical characteristics (e.g., size, type, distance from desti-

any *redundant* links each of whose congestion causes severe disconnection of others (e.g., links used *in series*).

[3]Power-law distributions arise naturally from different types of cost minimizations in many other fields. For example, research in linguistics clearly shows that power laws defining the frequency of word occurrences in random English text arise from the minimization of human-communication cost [30, 51].

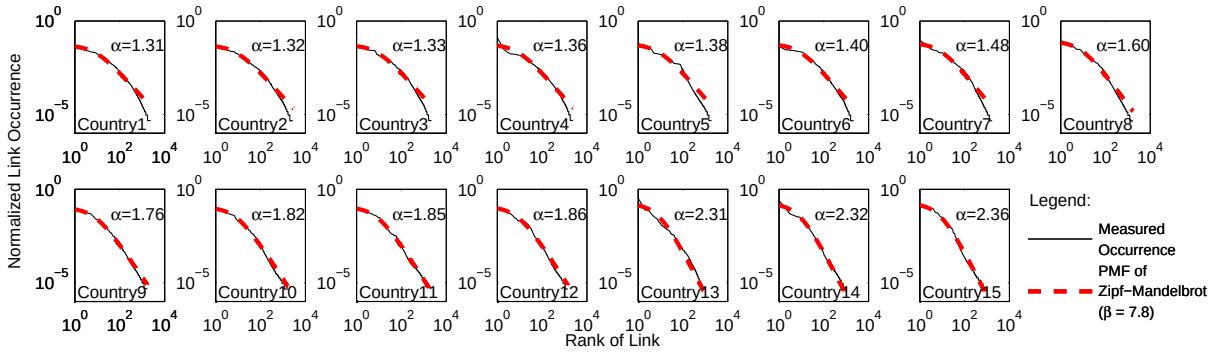

Figure 2: Normalized link occurrence/rank in traced routes to 1,000 randomly selected hosts in 15 countries.

nation hosts) as well as countermeasures against flooding attacks that attempt to exploit them. To illustrate the skew of link-occurrence distributions, we present our measurements for 15 countries and cities around the world in Fig. 2 and Fig. 14 (Appendix A), respectively. In these figures, we illustrate the relation between the link occurrence normalized by the total number of measured routes and the rank of links in *log-log* scale, for 1,000 servers in each country and city. The normalized occurrence of a link is the portion of routes between S and D carried by the link; e.g., if a link carries 10% of routes between S and D, its normalized occurrence is 0.10. We observe that the normalized link-occurrence distribution is accurately modeled by *Zipf-Mandelbrot* distribution; namely,

$$f(k) \sim 1/(k+\beta)^{\alpha}, \tag{1}$$

where k is the rank of the link, α is the exponent of the power-law distribution, and β is the fitting parameter. Exponent α is a good measure of route concentration, or distribution skew, and hence of bottleneck size: the higher α, the sharper concentration of routes in few links. Fitting parameter β captures the flatter values in the high-rank region (i.e., lower values on the x-axis). This region is not modeled as well by an ordinary Zipf distribution since its probability mass function would be a straight line in *log-log* scale on the entire range. The phenomenon of *flatter* occurrence in high-rank region is due to the nature of link sampling via route measurement; that is, multiple links are sampled together when each route is measured and there exist no duplicate link samples in a route in general due to the loop-freeness property of Internet routes. Thus, the occurrence of extremely popular links are limited.[4]

To enable comparison of route concentration in a few links of different destination regions, we fix the fitting parameter β and find the values of exponent α for the best fit across the fifteen countries; i.e., $\beta = 7.8$ causes the smallest fitting error. In Fig. 2 and Fig. 14 (Appendix A), the fifteen countries and cities are ordered by increasing value of α in the range 1.31 – 2.36.

2.3 Causes

What causes routing bottlenecks, or high skew/power-law distribution of link occurrence? Often, power-law distributions (especially Zipf-Mandelbrot distribution) arise from processes that involve some cost minimization; e.g., minimization of human-communication cost [30, 51]. Thus, one would naturally expect that power-laws in link-occurrence distributions are caused by the *cost minimization* criteria used for route selection and network design in the Internet; i.e., both intra- and inter-domain interconnections and routing in the Internet. Extra cost minimization is provided by the "hot-potato" routing between domains.

[4]The same phenomenon was explained and validated by the similar characteristics of data samples in [18].

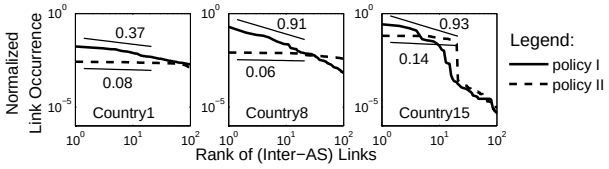

Figure 3: Normalized link occurrence/rank in simulated inter-AS links in three countries.

2.3.1 Cost minimization in inter-domain routing

Inter-domain routing policy creates routing bottlenecks in inter-AS links: BGP is the *de facto* routing protocol for inter-domain (i.e., AS-level) Internet connectivity. The *rule-of-thumb* BGP policy for choosing inter-AS paths is the minimization the network's operating cost; i.e., whenever several AS paths to a destination are found, the minimum-cost path[5] is selected. This policy is intended to minimize operating costs of routing in the Internet [15, 17].

To determine whether the rule-of-thumb routing policy contributes to the creation of routing bottlenecks, we run AS-level simulations on the CAIDA's AS topology[6] and compare the operation of the rule-of-thumb routing policy (i.e., policy I) and a *hypothetical* routing policy that distributes routes uniformly across possible inter-domain links (i.e., policy II). This hypothetical routing policy favors inter-domain links that serve fewer AS paths for a particular destination. We simulate this policy by Dijkstra's algorithm with dynamically changing link weight, which are proportional to the number of BGP paths served.

Fig. 3 shows the normalized link occurrence/rank plots for inter-AS links when we create BGP paths from all stub ASes to the ASes in *Country1*, *Country8*, and *Country15* ac-

[5]If there exist multiple same cost paths, the shortest path is selected.

[6]We use the dataset available for June 2012 from `http://www.caida.org/data/active/as-relationships/`

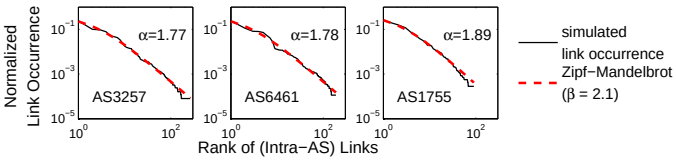

Figure 4: Normalized link occurrence/rank in simulated AS-internal routes for three ISPs.

cording to the two BGP policies. To clearly see the different skew of the link occurrence distribution of the two policies, we measure the slopes of link distributions in *log-log* scale in the high-rank region.[7] *Country1* and *Country8* have barely observable skew in this region (i.e., slope less than 0.1) with policy II while they have much higher skew (i.e., slope of 0.37 – 0.91) with policy I. *Country15* has a small skew (i.e., slope of 0.14) with policy II and a much higher skew of 0.93 with policy I. This suggests that, even though inter-domain Internet routes may have no physical bottlenecks (or very few, as in Country15), the BGP cost-minimization policy creates inter-domain routing bottlenecks.

2.3.2 Cost minimization in intra-domain network topology and routing

Internal AS router-level topology creates intra-domain routing bottlenecks: Most ISPs build and manage hierarchical internal network structures for cost minimization [42, 27] and these structures inherently create routing bottlenecks within ISPs. An ISP is composed of multiple points of presence (or POPs) in different geographic locations and they are connected via few high-capacity *backbone* links. Within each POP, many low-to-mid capacity *access* links connect the backbone routers to the border routers.

In general, ISPs aim to minimize the number of expensive long-distance high-capacity backbone links by multiplexing as much traffic as possible at the few backbone links [27]. As a result, backbone links become routing bottlenecks. To show this, we carry out simulations using Tier-1 ISP topologies inferred by Rocketfuel [42]. We construct ingress-egress routes for all possible pairs of access routers using *shortest-path routing* [29]. Fig. 4 shows the simulated normalized link occurrence/rank for the three ASes belonging to different ISPs. In all three ASes, we find that the Zipf-Mandelbrot distribution fits accurately with high value of α (i.e., 1.77 – 1.89) when β is deliberately fixed to 2.10 for best fitting and direct skew comparison; that is, a few AS internal links are extremely heavily used whereas most other internal links are very lightly used. Moreover, most of the heavily used links (i.e., 70%, 70%, and 90% of 10 most heavily used links in each of the three ISPs, respectively) are indeed backbone links that connect distant POPs. We reconfirm this later in Section 2.4.1 where we find that a large percentage (i.e., 30%) of links in routing bottlenecks are intra-AS links.

Hot-potato routing policy in ISPs aggravates inter-domain routing bottlenecks: The *hot-potato* routing policy is another example of a cost-minimization policy used by ISPs; i.e., this policy chooses the closest egress router among multiple egress routers to the next-hop AS [47]. As already reported [50], this policy causes a load imbalance at multiple inter-AS links connecting two ASes and thus aggravates the routing bottlenecks at the inter-AS links.

[7]Since the link-occurrence distribution of policy II is not modeled by Zipf-Mandelbrot distribution, we simply measure the slope in the high rank region to compare the skew.

2.4 Characteristics

In this subsection we investigate the characteristics of the links in the routing bottlenecks in terms of link types (e.g., intra-AS links, inter-AS links, or IXP-connecting links) and distance from the hosts in the target region (e.g., average router hops) as a backdrop to the design of countermeasures against attacks that exploit bottleneck links. The variety of link types found and their distribution make it *impossible* to design a single 'one-size-fits-all' countermeasure. Instead, in Section 5, we discuss several practical countermeasures that account for the specifics bottleneck link types.

2.4.1 Link types

We first categorize the three link types based on their roles in the Internet topology: intra-AS links, which connect two routers owned by the same AS, inter-AS links, which connect routers in two different ASes, and IXP-connecting links, which connect to routers in IXPs. Although the link types are clearly distinguished in the above definitions, the inference of link types via *traceroute* is known to be surprisingly difficult and error prone due to potential inference ambiguity [32]. For example, the *AS boundary ambiguity* [32] arises because routers at AS boundaries sometimes use IPs borrowed from their neighbor ASes for their interfaces. This is possible because the IPs at the both ends of the inter-AS links are in the same prefix. Borrowed IPs make it difficult to determine whether a link is an intra- or inter-AS link.

Our method of determining link type eliminates the AS boundary ambiguity by utilizing route diversity at the bottleneck links. Unlike the previous analysis [32, 21], our experiment measures a large number of disjoint incoming/outgoing routes to/from a bottleneck link. In other words, we gather all visible links 1-hop before/after the bottleneck link, and this additional information helps us infer the link types at AS boundary without much ambiguity.[8]

Fig. 5 summarizes the percentage of the link types of the 50 most occurred links for each of the 15 countries. The average percentage of all 15 countries is presented in the rightmost bar. Notice that the intra-AS and the inter-AS links are further categorized by the AS types, i.e., Tier-1, Tier-2, and Tier-3 ASes. The list of Tier-1 ASes are obtained from the 13 selected ASes in Renesys' Baker's Dozen[9]; Tier-3 ASes are the ones that have no customer but only providers or peers; and the rest of the ASes are labeled as Tier-2 ASes.

Our investigation found three interesting results. The first is that the link types are approximately evenly distributed. On average (see the rightmost bar in Fig. 5) 30% of them are intra-AS links, 30% are inter-AS links, and 20% are IXP-connecting links. The rest of 20% is not determined due to lack of *traceroute* visibility. We note that the intra-AS bottleneck links are located either in Tier-1 or Tier-2 ASes, but never in Tier-3 ASes. This high percentage of intra-AS bottleneck links in Tier-1 or Tier-2 ASes contradicts the common belief that large ISPs distribute routes over their internal links very well using complete knowledge

[8]For IP to ASN mapping, we utilize a public IP-to-ASN mapping database by Cymru (`https://www.team-cymru.org/Services/ip-to-asn.html`).

[9]`http://www.renesys.com/2014/01/bakers-dozen-2013-edition/`

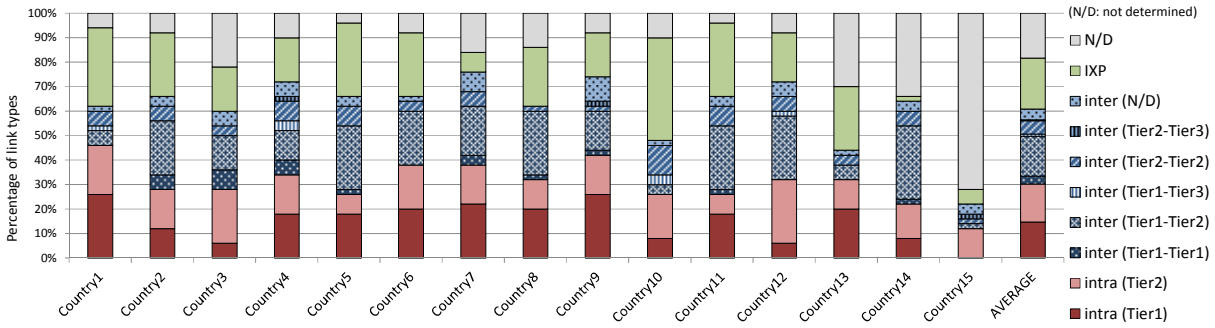

Figure 5: Percentage of link types of the 50 most occurred links for each of the 15 countries. Three link types (i.e., intra-AS links, inter-AS links, and IXP-connecting links) and three AS types (i.e., Tier-1, Tier-2, and Tier-3) are used for categorization.

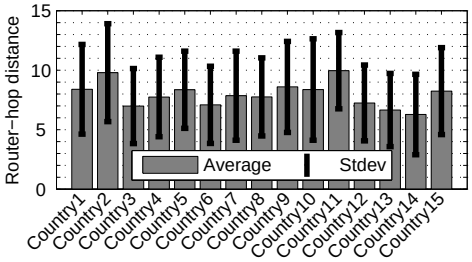

Figure 6: Router-hop distance of 50 bottleneck links for each of the 15 countries from the target regions.

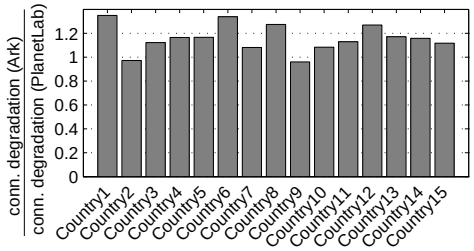

Figure 7: Connectivity degradation in the Ark dataset relative to the PlanetLab dataset for 50 flooding links selected from the routes measured by the PlanetLab nodes.

of, and control over, their own networks. This unexpected result implies that, in practice, large ISPs (both Tier-1 and Tier-2) are responsible for a large portion of the bottlenecks. The second interesting observation is that inter-AS links are also a common type of bottleneck links and the majority of these are connecting Tier-1 ASes. Again, we conclude that the Tier-1 ASes are responsible for the majority of inter-AS bottleneck links as well. The third interesting observation is that IXPs are another very popular type bottleneck links. This result is consistent with the recent trend of increasing popularity of IXPs [2].

2.4.2 *Link distance*

We also measure the *router-hop distance* of the bottleneck links from the hosts in the target regions. To measure a bottleneck link's distance, we take the average of router-hop distances from the multiple hosts in the region. One challenge in measuring the router-hop distance via *traceroute* is that more than half of the destinations used have firewalls in their local networks, which prevents discovery of the last few router hops from the destinations. When *traceroute* does not reach a destination we assume the presence of an invisible firewall that is directly followed by the destination. Thus, our distance is a strict *lower-bound* of the real distance from destination hosts.

Fig. 6 shows the average and standard deviation of the link distance of the 50 bottleneck links for each of the 15 countries. The average distance ranges from 6 to 10 router hops with average of 7.9 hops and no significant differences were found across the 15 countries. Considering the average length of Internet routes is approximately 17 router hops [12], we conclude that the bottleneck links are located in the middle to the slightly closer to the target region on the routes to the target. The distance analysis is also consistent with the observation that the most bottleneck links are within or connecting Tier-1 ASes.

3. ACCURACY OF BOTTLENECK MEASUREMENTS

3.1 Independence of Route Sources

One of the common pitfalls in Internet measurements is the dependency on vantage point; that is, the location where a measurement is performed can significantly affect the interpretation of the measurement [38]. Here we argue that our routing-bottleneck results are independent of the selection of route sources S. To show this, we validate our computation of routing-bottleneck results by comparing the connectivity degradation[10] calculated using the original source set S (i.e., 250 PlanetLab nodes) with that calculated using an *independent* source set S' (i.e., 86 Ark monitors),[11] as shown in Fig. 7. Notice that we select 50 bottleneck links for each country by analyzing the routes measured by Plan-

[10]Connectivity degradation is measured in terms of degradation ratio, which will be explained in detail in Section 4.2.

[11]The CAIDA's Ark project uses 86 monitors distributed over 81 cities in 37 countries and performs *traceroute* to all routed /24's. For consistent comparison, we use the Ark dataset that was measured on the same day when PlanetLab dataset was obtained and select a subset of the measured traces in the Ark dataset that has the same AS distribution of hosts used in the PlanetLab dataset.

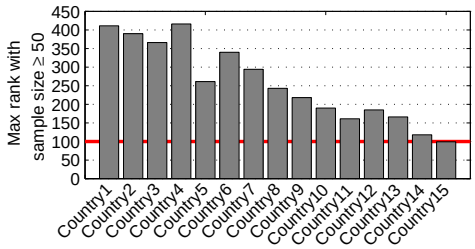

Figure 8: Maximum rank with sample size $\geq$ 50 for each of the 15 countries.

etLab nodes for *both* cases.[12] In most countries, the ratios of the two connectivity degradations are slightly higher or very close to 1, which means that the bottlenecks of the PlanetLab dataset also become the bottlenecks of the independent Ark dataset. This result confirms the independence of the bottleneck-link flooding results of the choice of route-source distribution [24].

3.2 Sufficiency of Link-Sample Size

Another common pitfall in Internet measurement discovering statistical properties of datasets is the lack of sample size; that is, it is possible that the sample size is not sufficiently large to detect possible deviations from a discovered distribution. For extracting reliable parameter estimates, the rule of thumb is that one needs to collect at least 50 samples for each value of element [8]. Fig. 8 shows the maximum rank of the links (ordered by decreasing link occurrence) that are observed with at least 50 link samples for the 15 countries in our measurement. The figure shows that for all 15 countries, all the high ranked links (i.e., rank $\leq$ 100) are observed with more than 50 link samples and thus the parameter estimates based on these links (i.e., α and β in Fig. 2) are statistically sound.

Fig. 9 confirms that we have collected a sufficient number of link samples. In this figure, we measure the normalized link occurrence with the various sizes of disjoint D and observe how the link occurrence in the high rank region (i.e., rank $\leq$ 100) converges. We can conclude that $|D| = 1{,}000$ is sufficient to discover the power-law distribution in rank $\leq$ 100 because it displays the same power-law distribution in the range as that observed with smaller size for D; i.e., $|D| < 1{,}000$. Thus, with a relatively small number of measurements one can learn the power-law distribution of the few but frequently observed high-rank links.

3.3 Traceroute Accuracy

traceroute is a very commonly used but frequently misused network monitoring tool. We discuss common pitfalls in analyzing *traceroute* results and describe how we avoid them.

Inaccurate alias resolution: In many previous 'topology measurement' studies, it is extremely important to accurately infer the group of interfaces located in the same router (or alias resolution) because its accuracy dramatically affects the resulting network topology [42, 34]. Highly accurate alias resolution still remains an open problem. Our measurements do *not* need alias resolution because we do not measure any router-level topology, but only layer-3 links (i.e., interfaces) and routes that use those links.

Inaccurate representation of load-balanced routes: Ordinary *traceroute* does not accurately capture load-balanced links and thus specially crafted *traceroute*-like tools (e.g., Paris traceroute [6]) are needed to discover these links. Our measurement does *not* need to discover load-balanced links because they cannot become the routing bottlenecks. Instead, we perform ordinary *traceroute* multiple times (e.g., 6 *traceroutes* in our measurement) for the same source-destination pair and ignore the links that do not always appear in multiple routes.

Inconsistent returned IPs: In response to *traceroute*, common router implementations return the address of the *incoming* interface where packets enter the router. However, very few router models return the *outgoing* interface used to forward ICMP messages back to the host launching *traceroute* [32, 33] and thus create measurement errors. However, our routing bottleneck measurement is not affected by this router behavior because (1) most of the identified router models that return outgoing interfaces are likely to be in small ASes since they are mostly Linux-based software routers or lower-end routers [32], and (2) we remove all load-balanced links that can be created by the routers which return outgoing interfaces [33].

Hidden links in MPLS tunnels: Some routers in MPLS tunnels might not respond to *traceroute* and this might cause serious measurement errors [54]. However, according to a recent measurement study in 2012 [10], in the current Internet, nearly all (i.e., more than 95%) links in MPLS tunnels are visible to *traceroute* since most current routers implement RFC4950 ICMP extension and/or ttl-propagate option to respond to *traceroute* [10].

4. ROUTING-BOTTLENECK EXPLOITS

Bottleneck links provide a very attractive target for link-flooding attacks [7, 24]. After identifying a routing bottleneck, an adversary chooses a set of links in it and floods them. In this section we discuss the selection of such links, the expected degradation in connectivity to the targeted hosts D, and the scaling property of the link-flooding attacks. To measure the strength of a link-flooding attack, we first define an ideal attack that *completely disconnects* all routes from sources S to selected hosts of destinations D. Then, we define lower-strength attacks that nevertheless *degrade* connectivity significantly.

4.1 Connectivity Disconnection Attacks

Let S be the 250 PlanetLab nodes and D the 1,000 randomly selected hosts in the target region; e.g., a country or a city. For efficient disconnection attacks, the adversary needs to flood only *non-redundant* links; that is, flooding of a link should disconnect routes that have *not* been disconnected by the other already flooded links. Link redundancy can be avoided by flooding the *mincut* of the routes from S to D, namely the *minimum* set of links whose removal disconnects *all* the routes from S to D, which is denoted by $M(S, D)$.

Finding $M(S, D)$ can be formulated as the *set cover problem*[13]: given a set of element $\mathcal{U} = \{1, 2, \cdots, m\}$ (called the

[12]For detailed strategy to select the links from routing bottlenecks, viz., Section 4.2.

[13]Notice that our *mincut* is *not* the same as a *graph-theoretic mincut*; our *mincut* is a set of links that cover all routes to chosen nodes whereas the *graph-theoretic mincut* is a set of *physical* link cuts for an arbitrary network partitioning.

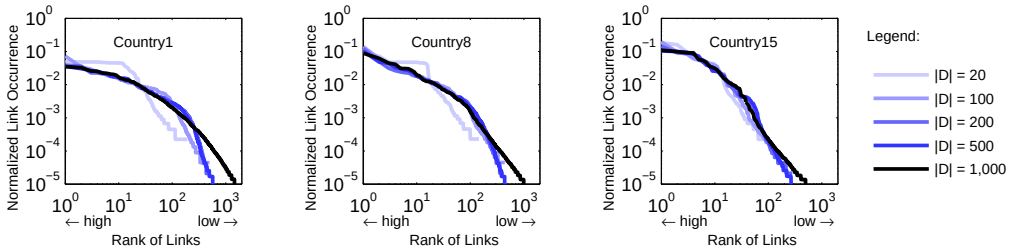

Figure 9: Normalized link occurrence/rank in traced routes to 1,000 randomly selected hosts in 3 countries.

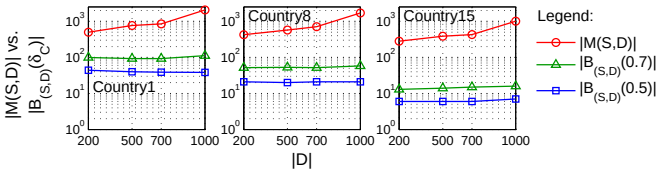

Figure 10: Measured sizes of mincuts, $M(S,D)$, and exploitable bottlenecks for given degradation ratios δ_C, $B_{(S,D)}(\delta_C)$, for varying size of D in 3 countries.

universe) and a set $\mathcal{K}$ of n sets whose union equals the universe, the problem is to identify the smallest subset of $\mathcal{K}$ whose union equals the universe. Thus, our *mincut* problem can be formulated as follows: the set of all routes we want to disconnect is the universe, $\mathcal{U}$; all IP-layer links are the sets in $\mathcal{K}$, each of which contains a subset of routes in $\mathcal{U}$, and their union equals $\mathcal{U}$; the problem is to find the smallest set of links whose union equals $\mathcal{U}$. Since the set cover problem is NP-hard, we run a greedy algorithm [19] to calculate $M(S,D)$. The greedy algorithm, which is similar to the one used to find critical links in the Crossfire attack [24], iteratively selects (and virtually cuts) each link in $M(S,D)$ until all the routes from S to D are disconnected. To be specific, at each iteration, the algorithm picks the most effective link within the not-yet-selected links.

Our experiments show that flooding an entire *mincut*, $M(S,D)$, in any of the fifteen countries and cities selected would be rather *unrealistic*; e.g., approximately 83 Tbps would be required to flood a *mincut* of 2,066 links with 40 Gbps link capacity for a flooding attack against 1,000 servers in *Country1*.[14] Worse yet, Fig. 10 (red line) shows that the *mincut* size, $|M(S,D)|$, grows as $|D|$ grows. This implies that any practical link-flooding attack that disconnects *all* the hosts of a target region will also be impractical because the *mincut* would likely be much larger than $M(S,D)$. However, as we show in the next section, an adversary does not need to flood an entire *mincut* to degrade connectivity of D hosts of a targeted region substantially.

4.2 Connectivity Degradation Attacks

Feasible yet powerful connectivity-degradation attacks would flood much smaller sets of links to achieve substantial connectivity degradation to the routes from S to D. To measure the strength of such attacks we define a connectivity-degradation metric, which we call the *degradation ratio*, as follows:

$$\delta_{(S,D)}(B) = \frac{\text{number of routes that traverse } B}{\text{number of routes from } S \text{ to } D}, \qquad (2)$$

where the *exploitable bottleneck* B is the set of links that are flooded by an attack.[15] B is a subset of the *mincut* $M(S,D)$, and its size, $|B|$, or the number of links to flood, is determined by an adversary's capability. Clearly, the maximum number of links that an adversary can flood is directly proportional to the maximum amount of traffic generated by attack sources controlled by the adversary; e.g., botnets or amplification servers. Here, we assume that the required bandwidth to flood a single link is 40 Gbps[16] and thus the adversary should create $40 \times n$ Gbps attack bandwidth to flood n links concurrently.

Fig. 11 shows the expected degradation ratio calculated for each of the 15 countries for varying number of links to flood, or $|B|$. These countries are ordered by increasing the averaged degradation ratio over $1 \leq |B| \leq 50$. By definition, the degradation ratio for B (i.e., $\delta_{(S,D)}(B)$) is the sum of normalized occurrences of the links in B. Thus, degradation ratio can be accurately modeled by the cumulative distribution function (CDF) of Zipf-Mandelbrot distribution since the normalized link occurrence follows Zipf-Mandelbrot distribution. Parameters α and β are listed in the plot. We observe that the ordering of the degradation ratio in Fig. 11 is exactly the same as the ordering of the values of α of the 15 countries in Fig. 2. That is, countries with low α (i.e., less skewed distribution) have low degradation ratio (i.e., less vulnerable to flooding attacks) and countries with high α (i.e., more skewed distribution) have high degradation ratio (i.e., more vulnerable to flooding attacks). This confirms that the skew of the link-occurrence distribution (or α in Zipf-Mandelbrot distribution) is a good indicator of the vulnerability of target regions to link-flooding attacks.

Fig. 11 also shows that the adversary can easily achieve significant degradation ratio (e.g., 40% - 82%) when flooding only few bottleneck links; e.g., 20 links. Given the proliferation of traffic amplification attacks achieving hundreds of Gbps or the extremely low costs of botnets, flooding several tens of bottleneck links of selected hosts in different countries around the world seems very practical. (We also illustrate the degradation ratio for 15 major cities in Fig. 15 in Appendix A.)

Thus, one cannot use well-known polynomial-time *mincut* algorithms in graph theory [48] for our purpose.

[14]Note that in calculating $M(S,D)$ we *exclude* network links that are directly connected to the hosts in S or D and thus $|M(S,D)|$ can be larger than $|S|$ or $|D|$.

[15]The definition of degradation ratio is similar to that presented in [24].

[16]Links with larger physical capacity (e.g., 100 Gbps) are recently introduced in the Internet backbone but the majority of backbone links are equipped with less than or equal to 40 Gbps of capacity [23].

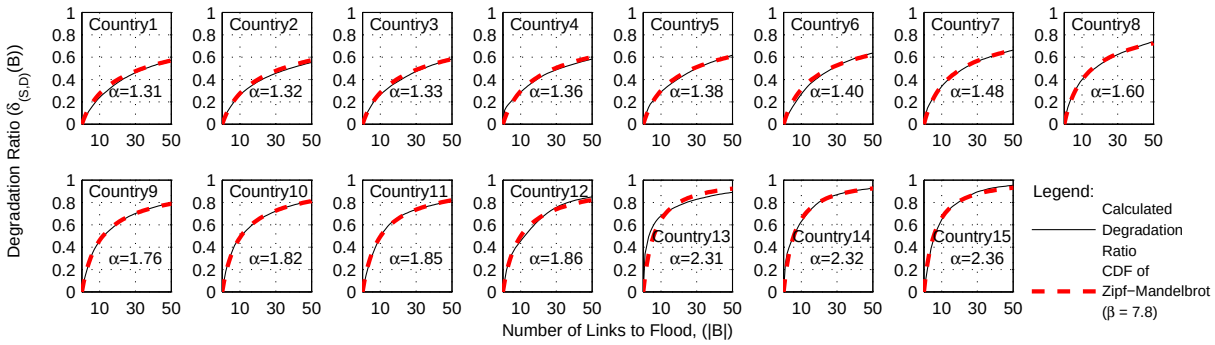

Figure 11: Calculated degradation-ratio/number-of-links-to-flood for 1,000 servers in 15 countries.

4.2.1 *Bottlenecks Sizes*

The size of an exploitable bottleneck selected for an attack clearly depends on the *chosen degradation ratio* δ_C sought by an adversary. This ratio is defined as:

$$B_{(S,D)}(\delta_C) = \text{minimum } |B|, \text{ such that } \delta_{(S,D)}(B) \geq \delta_C. \quad (3)$$

Exploitable bottlenecks, $B_{(S,D)}(\delta_C)$, are substantially smaller than their corresponding *mincuts*, $M(S,D)$. Fig. 10 shows the set sizes of the *mincuts* and the exploitable bottlenecks for given δ_C of 0.7 and 0.5 for varying sizes of D. The plots in the three countries show that $|M(S,D)|$ is one to two *orders of magnitude* larger than $|B_{(S,D)}(\delta_C)|$ in the entire range of measured $|D|$ and δ_C. In other words, the attack against the exploitable bottlenecks requires a much lower adversary's flooding capability while achieving substantial connectivity degradation (e.g., 70%). Figures 11 and 15 (Appendix A) illustrate the bottleneck sizes for different degradation ratios in 15 countries and cities around the world.

4.2.2 *Scaling the Attack Targets*

Our experiments suggest that an adversary needs not scale routing measurements and attack traffic much beyond those illustrated in this paper for much larger target-host sets (i.e., $|D| \gg 1,000$) in a chosen region to obtain connectivity-degradation ratios in the range illustrated in this paper. This is the case following two reasons. First, our measurements for multiple disjoint sets of selected hosts in a target region yield the *same* power-law distribution for different unrelated sizes of D; viz., Fig. 9. Hence, increasing the number of routes from S to a much larger D will not increase the size of the bottlenecks appreciably. In fact, we have already noted that, unlike the size of *mincuts*, $|M(S,D)|$, the size of the observed bottlenecks for a chosen degradation ratio δ_C, $|B_{(S,D)}(\delta_C)|$, does *not* change as $|D|$ increases, as shown in Fig. 10. Second, we have shown the routing-bottleneck discovery is independent of the choice of S, where $|S| \gg |B|$; viz., Section 3.1. This implies that, to flood the few additional bottleneck links necessary for a much increased target set D, an adversary needs not increase the size of S and attack traffic appreciably.

5. COUNTERMEASURES

Defenses against attacks exploiting routing bottlenecks range from some intuitive but naïve approaches, which will not work well, to some which will, namely structural countermeasures and operational countermeasures. We summarize these countermeasures, discuss their deployment challenges, and briefly evaluate their effectiveness. Naturally, defense mechanisms for *server-flooding* attacks (viz., [16]) are irrelevant to this disussion.

5.1 Naïve Approaches

The naïve approaches presented here are the most probable responses that the current networks would perform once the degradation attacks hit the routing bottleneck of any target region.

Local rerouting: Targeted networks can reactively change routes crossing flooded links so that the flooding traffic (including both legitimate and attack flows that are indistinguishable form legitimete flows) is distributed over multiple other local links. However, this might cause more collateral damage on the other local links after all.

Traffic-intensity based flow filtering: Typical mitigations for volumetric DDoS attacks detect and filter long-lived large flows *only* because otherwise they cannot run in real-time in large networks [26]. This countermeasure cannot detect nor filter attack flows in bottleneck links because these could be low-rate and thus indistinguishable from legitimate.

Using backup links: Typical backbone links are protected by the backup links, such that whenever links are cut, the backup links seamlessly continue to convey traffic. However, backup links cannot counter link-flooding attacks because they could be flooded too.

5.2 Structural Countermeasures

Structural countermeasures range from changes of physical Internet topology to those of inter-AS relationships. Although this type of countermeasures might have practical (e.g., business or legal) limitations, it could widen routing bottlenecks significantly. For example, if a country is connected to the rest of the world via only a handful of market-dominating ISPs, no matter how well routes are distributed, the country would inevitably experience routing bottlenecks. To widen these bottlenecks, the country would have to increase its route diversity to the outside world.

Fig. 12 illustrates the proposition that structural changes could be a fundamental solution to the routing bottleneck problem. The x-axis is the metric called *AS-level route diversity* and it is calculated as

$$\frac{\{\text{number of intermediate ASes}\}}{\{\text{average AS hops from Tier-1 ASes to target region}\}}, \quad (4)$$

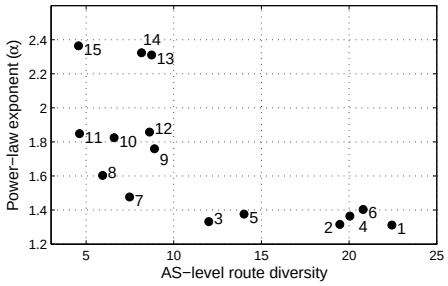

Figure 12: Correlation between power-law exponent and AS-level route diversity in 15 countries. (Legend: number i = Country$i$, for $i = 1, \cdots, 15$)

where the intermediate ASes are the ASes that connect the Tier-1 ASes with ASes located within each target region.[17] Ratio (4) is a good proxy for measuring the AS-level route diversity because it represents the average number of possible ASes at every AS hop from the Tier-1 ASes to the target region. The y-axis is the power-law exponent α obtained in Section 4.2. We see the clear correlation between the AS-level route diversity and the power-law exponent for the 15 countries, which supports our claim that the more AS-route diversity, the lower the power-law exponent. Moreover, we find the average AS-level route diversity of the seven European countries (i.e., 15.9) is more than two times larger than that of the rest of the countries (i.e., 7.7). We believe that this is because the market for ISPs in Europe is more competitive and thus providing more ISP choices for the Internet connectivity [14].

There exist various non-technical ways to increase the AS-level route diversity for a target region; e.g., regional regulatory authorities could encourage market entry by new ISPs; addition of new peerings to promote competition, or introduce new regulatory regimes that promote route diversity.

5.3 Operational Countermeasures

Operational countermeasures could improve the management plane of various routing protocols (e.g., BGP or OSPF) to either decrease the skew of link-occurrence distribution or better react to the exploits. Since this type of countermeasures does not change the network structure, it can be much easily implemented in practice.

5.3.1 Inter-domain load balancing

As seen in Section 2.4.1, about 30% of the bottleneck links are inter-AS links. When an inter-AS link is being flooded, at least one of the ASes should be able to quickly redirect the flooding traffic to relieve the severe congestion. However, a typical inter-AS inbound traffic engineering via tweaking BGP announcement [22] would be too slow due to the long BGP convergence time (e.g., 30 minutes [31]). Moreover, the outbound inter-AS level redirection can only be manually configured because inter-AS links are selected by the coupling of BGP and IGP (e.g., OSPF) protocols [47]. This coupling is (re)configured by human operators when it becomes necessary to diffuse flooding traffic [50].

In principle, an automated mechanism is needed to adaptively utilize multiple parallel inter-AS links [50] and/or multiple AS-level route with different next-hop ASes [55]. The specific design of such mechanisms is beyond the scope of this paper.

[17]The list of ASes within a target region are obtained from `http://www.nirsoft.net/countryip/`.

5.3.2 Intra-domain load balancing

Intra-domain load balancing can be an effective countermeasure because any balanced link *cannot* be the bottleneck.[18] Today's networks (especially, large ISPs) deploy intra-domain load-balancing mechanisms, such as Equal-Cost Multi-Path (ECMP) algorithm [20] and thus approximately 40% of Internet routes experience load balancing [6].

However, contrary to our expectation, we found that about 30% of the bottleneck links are located *within* Tier-1 or Tier-2 networks; viz., Section 2.4.1. This insufficient load balancing is caused by the inherent limitation of ECMP. That is, ECMP cannot balance all links in the network because it would be impossible to assign link weights in such a way that all routes have at least one alternate equal-cost route. Thus, ECMP alone cannot completely remove the potential bottleneck links in the networks.

To prevent the degradation attacks from targeting their internal links, large ISPs should identify commonly used but not load-balanced links and dynamically reconfigure their networks (e.g., by updating link weights) so that the identified links are load-balanced with other links.

5.3.3 Automated intra-domain traffic engineering

One of the most widely used traffic engineering mechanisms is MPLS. As of 2013, at least 30% of Internet routes travel through MPLS tunnels [10] and they are mostly deployed in the large ISPs. Unlike the local rerouting solution discussed in Section 5.1, MPLS reconfiguration can perform fine-grained traffic steering to avoid collateral damage on the other links.

However, the widely used offline MPLS reconfiguration cannot be very effective since it can reconfigure tunnels only on a time scale ranging from tens of minutes to hours and days [13, 52]. Worse yet, the online MPLS reconfiguration, called the auto-bandwidth mechanism [36], which automatically allocates required bandwidth to each tunnel and change routes, is susceptible to sustained congestion since it *cannot* detect congestion *directly* but only via reduced traffic rates caused by congestion. Thus, even auto-bandwidth mechanism would require human intervention to detect link-flooding attacks [44] thereby slowing reaction time considerably. Therefore, large ISPs need an automated control system that monitors link congestion and quickly reconfigures the related MPLS tunnels to be steered through other underutilized links.

5.3.4 Effectiveness of operational countermeasures

We evaluate the reduction of degradation ratios due to the following four defense strategies using the operational countermeasures: (1) inter-domain load balancing at all inter-AS links, (2) intra-domain load-balancing and traffic engineering at all intra-AS links, (3) all operational countermeasures at all Tier-1 ASes, and (4) all operational countermeasures at all Tier-1 and Tier-2 ASes. Fig. 13 shows the reduction of degradation ratios in percentage for 15 countries.[19] It shows that the defense strategies that protect a specific

[18]Notice that we remove any load-balanced links from our *traceroute* dataset for this reason (viz., Section 2.1).

[19]We do not flood a link when its link type is not determined.

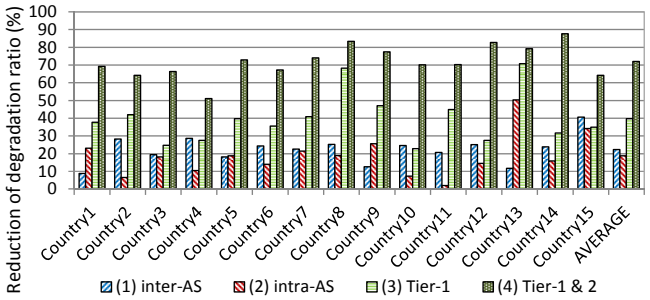

Figure 13: Reduction of degradation ratios due to four defense strategies when 20 bottleneck links are flooded for each of the 15 country.

type of links (i.e., strategy (1) and (2)) are not very effective in general (approximately 20% reduction on average) because adversaries can still find bottleneck links from the other types of links. However, the defense strategies deployed by Tier-1 and/or Tier-2 ASes (i.e., strategy (3) and (4)) show much higher effectiveness: when all Tier-1 ASes implement all the operational countermeasures, the degradation ratio is reduced by 40%; and when all Tier-2 ASes also join the defense, 72% of reduction is achieved on the average. This confirms our previous observation that the large Tier-1 and Tier-2 ASes are mainly responsible for the routing bottlenecks.

5.4 Application Server Distribution

One might distribute application servers in different geographic locations (possibly using content distribution networks, e.g., Akamai [37]) to distribute routes. The application servers have to be distributed in such a way that inherent route diversity is fully utilized; i.e., analysis must show that no routing bottleneck arise. However, this might not be practical for some domains such as industrial process systems, financial services, or defense services where constrained geography may restrict application distributions. These constraints may also limit the use of overlay networks [5] and services [25].

6. RELATED WORK

6.1 Internet Topology Studies

A large volume of research investigates the topology of Internet. Two long-term projects have measured the router-level Internet topology via *traceroute*-like tools: CAIDA's Archipelago project [1] and DIMES project [41]. Rocketfuel [42] is another project that use approximately 800 vantage points for *traceroute* to infer major ISP's internal topology. Together these studies provide important insights into the layer-3 topology of the Internet.

Our routing bottleneck measurement differs from the topology studies in two important ways. First, we do not measure or even infer the router-level topology but simply observe *how the routes are distributed* on the underlying router-level topology. Second, we do not need nor attempt to observe all the routes covering the entire address space but focus on the route-destination regions of potential adversary interest.

6.2 Topological Connectivity Attacks

Faloutsos *et al.* analyzed a massive amount of *traceroute* data and concluded that the node degree of the routers and ASes have power-law distribution [11]. Albert *et al.* confirmed the power-law behavior of the node-degree distribution and concluded that the Internet suffers from an 'Achilles' heel' problem; i.e., targeted removal attacks against the small number of hub nodes with high node degree will break up the entire Internet into small isolated pieces [4].

The Achilles' heel argument has triggered several counter-arguments. Some find that node-removal attacks are unrealistic because the number of required nodes to be removed is impractically high [28, 53]. Li *et al.* argue that the power-law behavior in node-degree distribution does not necessarily imply the existence of hub nodes in the Internet by showing that power-law node-degree distribution can be generated without hub nodes [27].

Our routing-bottleneck study discovers a new power-law distribution in the Internet. However, this power-law is *completely different* from that of the above-mentioned work for two reasons. First, we measure a power law for the link usage in Internet routes whereas the above-mentioned work finds power laws in the node-degree distribution. Second, the scope of out our power-law analysis is different; i.e., it is focused on, and limited to, a *chosen* route-destination region whereas the above-mentioned work analyzes the power-law characteristics of the entire Internet.

6.3 Bandwidth Bottleneck Studies

In the networking research, the term 'bottleneck' has been traditionally used to represent the link with the smallest available bandwidth on a route, i.e., the link that determines the end-to-end route throughput. To distinguish it from a routing bottleneck, we call this link the *bandwidth* bottleneck. Several attempts have been made to measure bandwidth bottlenecks in the Internet; viz., BFind [3] and PathNeck [21].

In contrast, routing bottlenecks are unrelated to the available bandwidth or provisioned link capacity, but closely related to the number of routes served by each link. In fact, the flooding attacks described in Section 4 turn routing bottleneck links into the bandwidth bottleneck links. In other words, the routing bottlenecks are *latent* bandwidth bottlenecks since they can always be exploited by an adversary and converted into bandwidth bottlenecks.

6.4 Control-Plane and Link-Flooding Attacks

Attacks that cause instability of the control plane in Internet routing [40] and link-flooding attacks [46, 24] have been recently proposed and launched in real life already [7]. Closest to our work, the Crossfire attack [24] presents the specific strategy to identify and flood a few targeted links for eight selected target areas in the US. In contrast to Crossfire, where the authors focus on the feasibility of flooding a small set of critical links, our study explores a *fundamental vulnerability* of today's Internet, namely, pervasive routing bottlenecks that can be exploited by any flooding attack. We show the ubiquity of routing bottlenecks in various countries and cities around the world via extensive measurements, and identify their basic cause. We also explore the characteristics of bottleneck links; e.g., link type and distance to targets. Lastly, we provide several practical countermeasures against bottleneck-link exploits, which also address all current attacks.

7. CONCLUSIONS

Routing bottlenecks, which arise naturally in the Internet, are defined by power-law distributions of link occurrence in routes to chosen destinations. Routing bottlenecks are pervasive, ubiquitous, and susceptible to scalable link-flooding attacks. Their prevention requires both specific structural and operational countermeasures whose deployment by network operators appears to be practical; i.e., it does not require major Internet redesign.

8. REFERENCES

[1] http://www.caida.org/data/active/as-relationships/.

[2] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger. Anatomy of a large european IXP. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, pages 163–174. ACM, 2012.

[3] A. Akella, S. Seshan, and A. Shaikh. An empirical evaluation of wide-area internet bottlenecks. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 101–114. ACM, 2003.

[4] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[5] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient Overlay Networks. In *Proceedings of the Eighteenth ACM Symposium on Operating Systems Principles*, SOSP '01, pages 131–145, New York, NY, USA, 2001. ACM.

[6] B. Augustin, T. Friedman, and R. Teixeira. Measuring load-balanced paths in the Internet. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 149–160. ACM, 2007.

[7] P. Bright. Can a DDoS break the Internet? Sure... just not all of it. In *Ars Technica*, April 2, 2013.

[8] A. Clauset, C. R. Shalizi, and M. E. Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

[9] DHS. Project Shine. *ICS-CERT Monitor, Quarterly Newsletter, Oct-Dec.*, pages 3–8, 2012.

[10] B. Donnet, M. Luckie, P. Mérindol, and J.-J. Pansiot. Revealing MPLS tunnels obscured from traceroute. *ACM SIGCOMM CCR*, 42(2):87–93, 2012.

[11] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *ACM SIGCOMM Computer Communication Review*, volume 29, pages 251–262. ACM, 1999.

[12] A. Fei, G. Pei, R. Liu, and L. Zhang. Measurements on delay and hop-count of the Internet. In *IEEE GLOBECOM.98-Internet Mini-Conference*, 1998.

[13] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving traffic demands for operational IP networks: methodology and experience. In *ACM SIGCOMM Computer Communication Review*, volume 30, pages 257–270. ACM, 2000.

[14] B. Fung. What Europe can teach us about keeping the Internet open and free. In *The Washington Post*, September 20, 2013.

[15] L. Gao. On inferring autonomous system relationships in the Internet. *IEEE/ACM Transactions on Networking (ToN)*, 9(6):733–745, 2001.

[16] M. Geva, A. Herzberg, and Y. Gev. Bandwidth Distributed Denial of Service: Attacks and Defenses. *IEEE Security & Privacy*, 12(1):54–61, January 2014.

[17] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How secure are secure interdomain routing protocols. In *ACM SIGCOMM CCR*, volume 40, pages 87–98. ACM, 2010.

[18] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In *ACM SIGOPS Operating Systems Review*, volume 37, pages 314–329. ACM, 2003.

[19] D. S. Hochbaum. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1996.

[20] C. E. Hopps. Analysis of an equal-cost multi-path algorithm. *RFC 2992*, 2000.

[21] N. Hu, L. E. Li, Z. M. Mao, P. Steenkiste, and J. Wang. Locating Internet bottlenecks: algorithms, measurements, and implications. In *Proceedings of SIGCOMM '04*, pages 41–54, New York, NY, USA, 2004. ACM.

[22] J. Hui Wang, D. M. Chiu, J. C. Lui, and R. K. Chang. Inter-as inbound traffic engineering via ASPP. *Network and Service Management, IEEE Transactions on*, 4(1):62–70, 2007.

[23] Internet2 Network – Layer3 / IP Connectors Map, 2013.

[24] M. S. Kang, S. B. Lee, and V. D. Gligor. The Crossfire Attack. In *IEEE Symposium on Security and Privacy*, pages 127–141. IEEE, 2013.

[25] A. D. Keromytis, V. Misra, and D. Rubenstein. SOS: secure overlay services. In *Proceedings of SIGCOMM '02*, pages 61–72, New York, NY, USA, 2002. ACM.

[26] R. Krishnan, M. Durrani, and P. Phaal. Real-time SDN Analytics for DDoS mitigation. *Open Networking Summit*, 2014.

[27] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the Internet's router-level topology. In *ACM SIGCOMM Computer Communication Review*, volume 34, pages 3–14. ACM, 2004.

[28] D. Magoni. Tearing down the Internet. *Selected Areas in Communications, IEEE Journal on*, 21(6), 2003.

[29] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson. Inferring link weights using end-to-end measurements. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 231–236. ACM, 2002.

[30] B. Mandelbrot. Information theory and psycholinguistics. *BB Wolman and E*, 1965.

[31] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz. Route flap damping exacerbates Internet routing convergence. In *ACM SIGCOMM Computer Communication Review*, volume 32, pages 221–233. ACM, 2002.

[32] Z. M. Mao, J. Rexford, J. Wang, and R. H. Katz. Towards an accurate AS-level traceroute tool. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 365–378. ACM, 2003.

[33] P. Marchetta, V. Persico, E. Katz-Bassett, and A. Pescapé. Don't trust traceroute (completely). In *ACM CoNEXT Student workshop*, 2013.

[34] P. Marchetta, V. Persico, and A. Pescapé. Pythia: yet another active probing technique for alias resolution. *ACM CoNEXT*, pages 229–234, 2013.

[35] M. Mimoso. 400 Gbps NTP Amplification attack alarmingly simple. In *Threatpost*, Feb. 13, 2014.

[36] MPLS Traffic Engineering (TE)–Automatic Bandwidth Adjustment for TE Tunnels. http://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/fsteaut.html#wp1015347.

[37] E. Nygren, R. K. Sitaraman, and J. Sun. The Akamai network: a platform for high-performance Internet applications. *ACM SIGOPS Operating Systems Review*, 44(3):2–19, 2010.

[38] V. Paxson. Strategies for sound Internet measurement. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 263–271. ACM, 2004.

[39] PlanetLab. http://www.planet-lab.org/.

[40] M. Schuchard, A. Mohaisen, D. Foo Kune, N. Hopper, Y. Kim, and E. Y. Vasserman. Losing control of the Internet: using the data plane to attack the control plane. In *Proceedings of NDSS 2011*, pages 726–728, New York, NY, USA, 2010. ACM.

[41] Y. Shavitt et al. The DIMES Project, 2008.

[42] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. *ACM SIGCOMM*

CCR, 32(4):133–145, 2002.
[43] R. Steenbergen. A practical guide to (correctly) troubleshooting with traceroute. *North American Network Operators Group*, pages 1–49, 2009.
[44] R. Steenbergen. MPLS RSVP-TE Auto-Bandwidth - Lessons Learned. *NANOG 58*, 2013.
[45] C. Strohm and E. Eric. Cyber Attacks on U.S. Banks Expose Computer Vulnerability. In *Bloomberg*, Sept. 27, 2012.
[46] A. Studer and A. Perrig. The Coremelt attack. In *Proceedings of ESORICS'09*, pages 37–52, Berlin, Heidelberg, 2009. Springer-Verlag.
[47] R. Teixeira, A. Shaikh, T. Griffin, and J. Rexford. Dynamics of hot-potato routing in IP networks. In *ACM SIGMETRICS Performance Evaluation Review*, volume 32, pages 307–319. ACM, 2004.
[48] H. Thomas et al. *Introduction to algorithms*. MIT press, 2009.
[49] Y. Tian, R. Dey, Y. Liu, and K. W. Ross. China's Internet: Topology mapping and geolocating. In *INFOCOM, 2012 Proceedings IEEE*, pages 2531–2535. IEEE, 2012.
[50] P. Verkaik, D. Pei, T. Scholl, A. Shaikh, A. C. Snoeren, and J. E. Van Der Merwe. Wresting Control from BGP: Scalable Fine-Grained Route Control. In *USENIX ATC*, pages 295–308, 2007.
[51] P. Vogt. Minimum cost and the emergence of the Zipf-Mandelbrot law. In *Artificial Life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*, 2004.
[52] N. Wang, K. Ho, G. Pavlou, and M. Howarth. An overview of routing optimization for Internet traffic engineering. *Communications Surveys Tutorials, IEEE*, 10(1):36 –56, quarter 2008.
[53] Y. Wang, S. Xiao, G. Xiao, X. Fu, and T. H. Cheng. Robustness of complex communication networks under link attacks. In *Proceedings of ICAIT '08*, pages 61:1–61:7, New York, NY, USA, 2008. ACM.
[54] W. Willinger, D. Alderson, and J. C. Doyle. *Mathematics and the internet: A source of enormous confusion and great potential*. Defense Technical Information Center, 2009.
[55] W. Xu and J. Rexford. MIRO: Multi-path Interdomain ROuting. In *SIGCOMM'06*, 2006.

APPENDIX

A. ROUTING BOTTLENECKS FOR 15 MAJOR CITIES

In this Appendix, we summarize the measurement results of the routing bottlenecks in 15 major cities around the world.

Fig. 14 shows the normalized link-occurrence distribution for {*City1*,$\cdots$,*City15*}. This list is a permutation of the following 15 cities that are alphabetically ordered: {Beijing, Berlin, Chicago, Guangzhou, Houston, London, Los Angeles, Moscow, New York , Paris, Philadelphia, Rome, Shanghai, Shenzhen, and Tianjin}. The 15 cities are labeled and ordered by increasing the skew of the occurrence distribution; i.e., α of Zipf-Mandelbrot distribution.

Fig. 15 shows the degradation ratio for the fifteen major cities. The cities are ordered by increasing the averaged degradation ratio over $1 \leq |B| \leq 50$. This order is exactly the same as that of Fig. 14, which suggests that the value of α is a good indicator of the vulnerability of the target cities to link-flooding attacks.

The average value of α of the five European cities (i.e., 1.57) is noticeably smaller than those of the five US cities

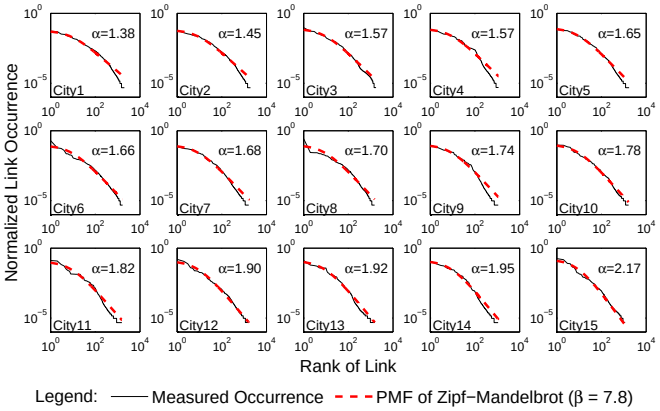

Figure 14: Normalized link occurrence/rank in traced routes to 1,000 randomly selected hosts in 15 cities.

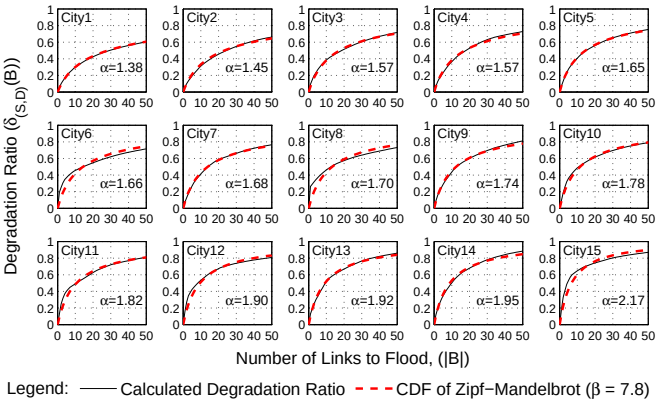

Figure 15: Calculated degradation-ratio/number-of-links-to-flood for 1,000 servers in 15 cities.

(i.e., 1.79) and the five Chinese cities (i.e., 1.82). This is due to the diversity of ISP choices in these regions; e.g., European cities have more choices of ISPs due to high competition and low cost of entry in their ISP markets while cities in the US and China have noticeably fewer choices [14, 49].