TowerDefense: Deployment Strategies for Battling against IP Prefix Hijacking

Tongqing Qiu*, Lusheng Ji†, Dan Pei†, Jia Wang† and Jun (Jim) Xu *
* College of Computing, Georgia Institute of Technology, Atlanta, GA
Email: {tongqqiu,jx}@cc.gatech.edu
†AT&T Labs – Research, Florham Park, NJ
Email: {lji, peidan, jiawang}@research.att.com

***Abstract*—IP prefix hijacking is one of the top security threats targeting today's Internet routing protocol. Several schemes have been proposed to either detect or mitigate prefix hijacking events. However, none of these approaches is adopted and deployed on a large-scale on the Internet for reasons such as scalability, economical practicality, or unrealistic assumptions about the collaborations among ISPs. Thus there are no actionable and deployable solutions for dealing with prefix hijacking.**

In this paper, we study key issues related to deploying and operating an IP prefix hijacking detection and mitigation system. Our contributions include (i) deployment strategies for hijacking detection and mitigation system (named as TowerDefense): a practical service model for prefix hijacking protection and effective algorithms for selecting agent locations for detecting and mitigating prefix hijacking attacks; and (ii) large scale experiments on PlanetLab and extensive analysis on the performance of TowerDefense.

I. Introduction

IP Prefix Hijacking attacks threaten the Internet's routing infrastructure. Such an attack exploits the inherent assumption of self-policing and trust among participants of BGP [1] protocol (the inter-domain routing protocol that is used on today's Internet) and injects false route announcements into the global routing infrastructure. These false route announcements, by using methods such as including non-existent AS links, make the attacker' Autonomous System (AS) appear attractive for forwarding data traffic destined for the victim IP address prefix. Lacking effective means for verifying the accuracy and authenticity of BGP route announcements, ASes that receive such false route announcements may accept and propagate the false route, as well as subsequently forward traffic destined for the victim prefix according to the false path. As a result, affected data traffic is diverted, or "hijacked", to ill-intentioned locations, causing performance degradation, service outage, and security breach for the victim prefix.

The importance of defending against IP prefix hijacking is well recognized by both industry and research communities, and many solutions [2]–[21] have been proposed in order to prevent, detect, locate, or mitigate IP prefix hijacking. These proposed solutions range from secure alternatives of BGP based routing protocol and infrastructure to techniques that detect ongoing attacks. While a more secure interdomain routing protocol would be ultimate solution for preventing prefix hijacking attacks, it usually requires adaptation and coordination from router vendors and ISPs. Hence, instead of waiting for a fix of the BGP protocol, deploying IP prefix hijacking defense systems on the Internet has been proposed as an immediate remedy. In this paper, we systematically study deployment related issues for IP prefix hijacking defense systems. We specifically address two key issues for an operational deployment of any prefix hijacking detection and mitigation system: (i) who should deploy and operate such a system, and (ii) how to deploy such kind of systems.

This paper also proposes two practical deployment strategies based on systematically studying prefix hijacking protection agent placements. The first is a new service model in which the service providers in particular the ISPs and CDN providers can deploy and operate a prefix hijacking detection and mitigation system for protecting their customers. The second deployment strategy includes two principles for protection agent placement, namely the *1. Detection principle*: to effectively detect a particular prefix hijacking attack, the detection system needs to have agents deployed in the region within which the routers are "polluted" with false route entries injected by the attack, and the *2. Mitigation principle*: to effectively mitigate a hijack, traffic to target prefix can be detoured towards pre-deployed relaying agents in order to avoid the polluted region of an prefix hijacking attack. We show in the paper that the agent location placement problem is NP, and propose effective greedy algorithms for it. Note that we focus on practical deployment strategies, rather than any new detection or mitigation methodologies. Moreover, we choose the simple greedy algorithm because the evaluation illustrates a decent result.

Because the problem of deploying and operating a prefix hijacking protection system is similar to that of a popular strategy computer game genre "Tower Defense" [22], appearing in best-selling game titles such as StarCraft, Age of Empires, and WarCraft, we name the aforementioned strategies as TowerDefense and the system built by following TowerDefense strategies as TowerDefense system. For the same reason, the deployed agents are sometimes called "towers" in this paper.

We conduct extensive analysis and large-scale experiments on PlanetLab to show that on a topology like today's Internet, TowerDefense deployed by a CDN or ISP provider can detect up to 99.8% and mitigate up to 98.2% of prefix hijacking attacks targeting at individual customers of the

978-1-4244-8645-8/10/$26.00 ©2010 IEEE

same provider with as few as 6 vantage points (i.e. where agents are deployed). To further highlight the practicality of TOWERDEFENSE we show through a case study of a Tier-1 ISP that (i) high detection/mitigation ratios can be achieved also through adding an even smaller number of new vantage points (which a service provider can obtain by buying transit from other ISPs) to the service provider's existing vantage point infrastructure, and (ii) even when 800 customers of the ISP sign up for the TOWERDEFENSE service, the total number of vantage points serving them remains small ($\sim$20).

The rest of the paper is organized as follows. Section II gives an overview of the TOWERDEFENSE strategies. Section III presents the detailed methodology for vantage point selections for detection and mitigation purpose. Then we analyze the selection results based on extensive simulations in Section IV. Section V evaluates the performance of TOWERDEFENSE on Planetlab. Section VI briefly surveys related works and we conclude in Section VII.

II. TOWERDEFENSE FRAMEWORK

A. Service Model

We believe that *protection against prefix hijacking* is most suitable to be offered by service providers in particular ISPs and CDN providers to their existing customers.

Firstly, since the protection service is provided by an entity that a customer is already buying other services (e.g. communications, content hosting, etc) from, the customer likely has more confidence and convenience to subscribe from them than from any new third parties.

Secondly a major issue in deploying a new service is cost. In this aspect, service providers are positioned far better than other potential parties because of their existing infrastructures. A CDN service provider may have already deployed its servers at a large number of locations ranging from dozens to thousands of ASes. All these locations can potentially be used as vantage points for prefix hijacking protection. For ISPs, firstly, a large ISP (e.g. tier-1 ISPs) may already own a few ASes spanning large geological area; secondly, an ISP is aware of the routes used by its neighboring ASes because its border routers have established BGP sessions with the neighbors; and thirdly, if the identified vantage point location (say AS T) is far away, an ISP can make up the capability simply purchasing a connectivity from AS T as a BGP customer and connect its prefix hijacking protection equipments (devices that run prefix hijacking detection and/or mitigation process) with the border router which runs BGP session with AS T. It is a simple and effective way to collaborate with other ISPs.

Moreover, although the service is offered for protecting customers of the service provider, in fact what gets protected are the inbound traffic paths towards the networks of these customers. If a hijacker can only hijack traffic from regions that has very little traffic for the target network, this hijacking is as good as non-effective. Thus knowing who communicate with the protected networks gives tremendous advantage for whoever offers the protection. This is exactly where ISPs and CDN providers have extensive knowledge.

B. Prefix Hijacking Protection

When a hijacker launches its hijacking attack against a target network, using a BGP router in its AS the hijacker spreads out false route announcements for the target prefix. Upon receiving such route announcements, some routers may accept the false routes and subsequently propagate to their neighbor routers while others may ignore such announcements. As a result, a portion of the Internet is *polluted* by the false routes announced by the hijacker. In the polluted region, routers now use the hijacker's false routes for forwarding packets addressed for the target prefix. In other words, any traffic that originates from or passes through the polluted region are now "hijacked".

Because typically only a portion of the Internet is polluted, an attack can only be detected if there are detection agents deployed in or right at the boundaries of the polluted region so that they can gather information regarding the false route for detecting anomalies. Typically, the agents comprise a cluster of machines for fault tolerance. Because the location and size of the polluted region of an attack vary depending on the locations of both the hijacker and target network, it is important to study where to place such detection agents to achieve optimum detection ratio for all possible hijacker locations.

Similarly, it is important to study where to deploy agents that may assist in mitigating prefix hijacking attacks. Different from mitigation approaches such as [12] which are aiming at correcting the false routes, we believe that a traffic redirection approach (e.g. IP tunneling and DNS-based redirection [23], [24]) may be more desirable because it can potentially react very rapidly. Also this approach can be applied by a wider range of providers, not only by those who are deeply vested in BGP operations.

For mitigating a hijack, there can actually be two types of redirections, which we refer to as *reflecting* and *mirroring*. When a reflector r is used in mitigation against a hijacking event on the target d, traffic from a source s destined to d will be re-routed to r and then from r to d. On the other hand when a mirror m is used in mitigation, traffic from s to d will be re-directed to m, and m will function as a mirroring site of d and respond to incoming traffic in the same way as d does.

An AS r can be used as a reflector site for s-d during a hijacking event only if both the path from s to r and the path from r to d are not polluted by the hijacking event. In addition, because the hijacker may know who the reflector r is, the path from s to r must not be polluted by hijacking event launched by the same hijacker on r either. On the other hand, the requirement for an AS m being used as a mirror for mitigating hijacking event on target d is that the path from s to m is not polluted by the hijacking event on d and the path from s to m is not polluted by the hijacking event on m. Although the requirement for a mirror site is more relaxed than reflector site, mirrors tend to be more expensive because they need to replicate contents. In addition, mirrors are better for less frequently changed contents.

Here again the key for a successful hijack mitigation service

is to place the mitigation agents, reflectors or mirrors, at strategically important locations so that they can mitigate the most attacks for the most sources of the target network. Hence, in this paper we mainly focus on placement strategy for detection and mitigation agents, which we call *towers*.

III. Methodology

In this section, we describe the methodology for a service provider to strategically select locations for its detection towers and mitigation towers to defend its customers against hijacking attacks. Because prefix hijacking is targeting inter-domain routing infrastructure, we consider ASes being the basic element.

Tower location selection involves evaluating many imaginary hijacking scenarios in the Internet AS topology, and assessing whether ASes may be impacted by the attacks. A service provider can infer Internet AS topology from publicly available BGP tables and updates such as Route Views [25] and RIPE [26]. We leave the discussion on the impact of the well-known topology incompleteness to Section IV-C. Moreover, the AS topology changes over time. The towers can be re-selected based on the same algorithm when the topology have significant changes. We do not consider any transit changes of AS topology, which can be handled by sophisticated detection algorithms like [18].

If an AS prefers a fake path to d announced by a hijacker h over the AS' current legitimate path to d, this AS is impacted/polluted by the hijacking. Subsequently not only will this AS propagate the fake path to its neighbors, which in turn determine if they prefer the fake path, any future traffic destined for d passing through the impacted AS is hijacked. In evaluating hijacking scenarios, the selection algorithm determine AS path preference based firstly on inter-domain routing policies, then preferring shorter AS path, and finally using random selection to break any remaining ties. Two widely adopted inter-domain routing policies are "prefer customer routes" and "valley-free routing" [27]. That is, while forwarding traffic an AS always prefers to forward using a link to its customer over a link to its peer over a link to its provider. Moreover, after traversing a provider-to-customer link or a peer link, a path will not traverse another customer-to-provider link or another peer link. Such kind of methodology is the state of art for understanding prefix hijacking problem [18], [19], [28].

We assume that only one hijacker AS involved into one prefix hijacking events. It is difficult to identify all attackers when multiple attackers advertise different false routes simultaneously.

A. Detection Tower Selection

TowerDefense can employ existing detection mechanism [2], [14], [15], [17], [18], [29] for detecting hijacking events. While the actual detection methods differ by these approaches, they generally require the presence of detection agents in impacted ASes, to collect data plane and/or control plan information. Thus to keep our evaluation method general, we assume that if the service provider has at least a detection tower deployed in one of the impacted ASes, the hijacking event can be detected.

Therefore the detection tower position selection problem can be formulated as the following. Given a customer prefix d and a set of candidate detection tower locations V_c, we need to find the minimum subset V_d of V_c that the detection towers v in V_d can detect as many as possible hijacking events targeting a customer prefix d. Obviously, the selection is per prefix based. If the candidate set contains all ASes on the Internet, the problem does become a classic *set cover* problem, which is NP hard. But in reality, the set of candidate locations is limited, and same for detection tower selection. Therefore, to select the detection tower is at least as hard as to solve the set cover. We adopt a greedy algorithm similar to that for set cover problem to solve this problem.

More specifically, the undetected hijacker AS set H_u was first initialized to all possible hijacker ASes set H for hijacking d and the selected detection tower set V_d is empty. In each iteration, we select a detection tower v from candidate set V_c that can detect the most hijackers H_v from the undetected set H_u and move it out of the candidate set V_c into the selected detection tower set V_d. At the same time, we update the set of undetected hijacker AS set H_u by taking out the hijacker ASes that v can detect. The selection process can be terminated either after a fixed number of detection towers are selected (up to all candidate ASes) or after the gain in the detection coverage by adding a new detection tower becomes marginal (e.g., below a given threshold). The algorithmic description of this algorithm is in [30]. More formally, we define *detection coverage* $DE(v,d)$ of a detection tower v against hijackers attacking d as $DE(v,d) = |H_v|/|H|$. Then the *detection coverage of a subset of detectors* V_d is:

$$\mathcal{DE} = |\bigcup_{v \in V_d} H_v|/|H|$$

The above greedy algorithm is to maximize the detection coverage.

B. Mitigation Tower Selection

Similar to detection tower selection, mitigation tower selection is a variant of set cover and can be done by a very similar greedy algorithm with one difference, the criteria for picking one candidate mitigation tower location over the others during each iteration. We first define *mitigation coverage* of a mitigation tower m against an *individual* hijacker h attacking d as the following:

$$\mathcal{ME}_I(h,m,d) = \frac{|MS(h,m,d)|}{|S(h,d)|},$$

where $S(h,d)$ is the set of d's sources whose traffic will be hijacked by h and $MS(h,m,d)$[1] is the subset of sources of

[1]Although we do not explicit distinguish reflectors from mirrors here, obviously in actual computation the $MS(h,m,d)$ of an mitigation AS used as a reflector will be different from that of as a mirror.

$S(h,d)$ that m can mitigate. Then we define the *mitigation coverage against a set of hijackers* of a mitigation tower as:

$$\mathcal{ME}_S(m,d) = \sum_H \mathcal{ME}_I(h,m,d)/|H|,$$

where H is the set of hijackers in question.

The mitigation tower selection algorithm, which tries to maximize the mitigation coverage, is now described as follows. Initially, the unmitigated hijacker AS set H equals to all possible hijacker ASes for hijacking d and the selected mitigation tower set M_d is empty. In each iteration, we select a mitigation tower m from candidate set M_c that has the highest mitigation coverage against hijackers in H and move it out of the candidate set M_c into the selected mitigation tower set M_d. At the same time, we update the mitigation coverage for each mitigation tower in remaining candidate set M_c. The selection process can be terminated either after a fixed number of mitigation towers selected or after the gain in the mitigation coverage by adding a new mitigation towers becomes marginal.

IV. Analyzing Effectiveness of TowerDefense

In this section we evaluate the effectiveness of the detection and mitigation selection methods proposed in Section III, by exhaustively simulating hijacking events on the AS level topology of the Internet with all possible locations of hijacker ASes and target ASes.

In our experiments, we construct the AS level topology graph using BGP tables and routing updates obtained from RouteViews and RIPE in 2008. The resulting AS topology has over 28K ASes. Using the method proposed in [31], we classify them into four tiers based on their relationships (e.g., provider, customer, or peer) to other ASes. There are 9 well-known Tier-1 ISPs, 221 Tier-2 ASes, 22856 stubs, which are the lowest tier ASes with only customer-to-provider links. The remaining 5794 ASes are *Others*, which are between Tier-2 and stubs in the hierarchy.

A. Detection Effectiveness

In this section we evaluate the detection effectiveness of a service provider who would like to offer TowerDefense service to its stub customers. Single-provider stubs and multi-provider stubs are analyzed separately because the former's results are easier to analyze. We run the detection tower selection algorithm presented in Section III for each TowerDefense service provider (X) and each of its stub customers as the target d. We compute the average detection coverage over d for each X, which is then averaged over service providers' locations in the AS hierarchy (Tier-1, Tier-2, and Others). In order to trade between number of detection towers selected and the detection coverage, the selection process is terminated after the gain in the detection coverage by adding a new detection tower becomes marginal (below 0.5%). Detection tower selection guidelines are further summarized based on these results.

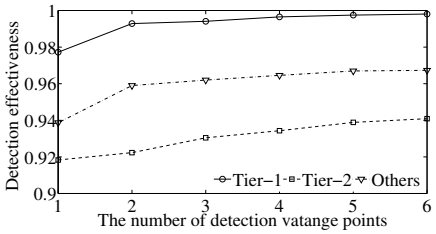

Fig. 1. Detection coverage for single-provider stubs as the number of detection towers increases.

1) Single-Provider Stubs: Figure 1 shows the average detection coverage in *Tier-1*, *Tier-2*, and *Others* when increasing the number of detection towers. We make the following observations. (i) The first selected detection tower can cover a very high percentage (e.g. more than 93% in Tier-1) of hijackers. (ii) The gain on the coverage by adding additional detection towers becomes marginal after a very small number of detection towers are selected. (iii) Tier-1 service providers achieve highest detection coverage (e.g. up to 99.8%). (iv) Tier-2 service providers achieve the lowest detection coverage. Detailed analysis are provided below.

Which AS is selected first as the detection tower? Our greedy algorithm chooses the AS with the best detection coverage as the first detection tower. We use real examples from our simulation traces to illustrate the insights behind such selections in Figure 2[2]. In Figure 2, there are three examples, one for a TowerDefense service provider at each tier: AS7018 for Tier-1, AS13249 for Tier-2, and AS2854 for Tier-3. The shaded node is the first detection tower selected by the greedy algorithm. d is one representative single-provider stub customer AS of the service provider X (the detection coverage of any other single-provider stub customer ASes of the same provider X is the same as that of d).

In Figure 2(a), AS3261 (a small ISP with some customers but only one provider AS35320) is chosen as the first detection tower for TowerDefense service provider Tier-1 AS7018. AS3261 can observe more than 96.1% of hijacking events targeted at d. This is mainly because its sole provider AS35320 (a Tier-2 AS) can be easily impacted by the hijacking event of target d, and then propagates the polluted path to AS3261. In addition, AS3261 can observe some other hijacking events if the attacker is a customer of AS3261, which AS35320 cannot observe.

Let us explain why AS35320 can be easily polluted now. AS35320 has two Tier-1 providers AS15097 and AS7459. It also peers with many (45) large Tier-1/Tier-2 ASes. Originally, AS35320 will choose the route AS35320 - AS7459 (or AS15097) - AS7018 to destination d. This original route is a provider route, which is less preferred than a peer route or a customer route, according to the BGP best path selection process. Therefore, AS35320 will be polluted if (i) the hijacker is Tier-1 provider of AS35320 (e.g. $h1$ in Figure 2) because

[2]Note that the figure play the role of illustration. We cannot directly choose ASes from this figure by hand

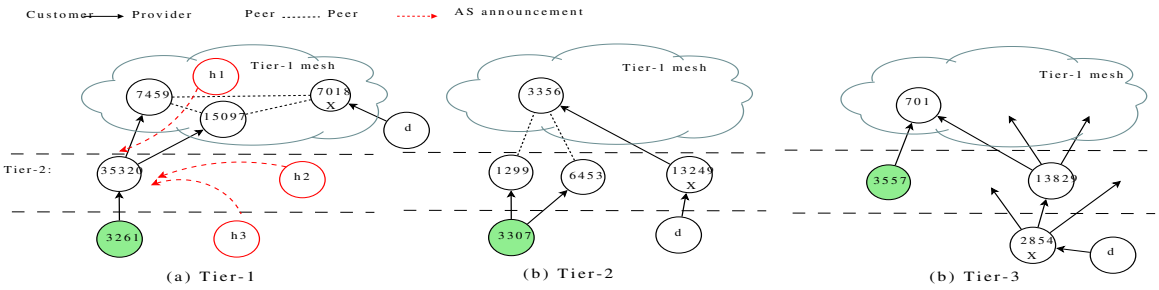

Fig. 2. Examples from simulation traces to explain tower selection for single-provider stub target d with provider at different locations.

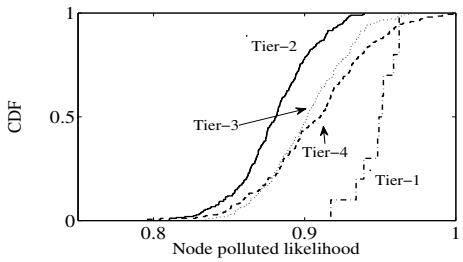

Fig. 3. Pollution likelihood for providers.

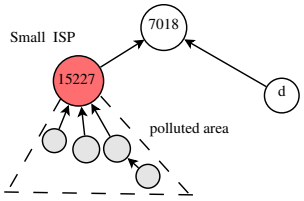

Fig. 4. Locally polluted example

the route AS35320-h1 is shorter than the original route to d; (ii) the hijacker is in its Tier-2 peers (e.g. $h2$ in Figure 2) because it prefers a peer route than a provider route; or (iii) the hijacker is in a lower-tier ASes (e.g. $h3$ in Figure 2) and the fake announcement reaches any of AS35320's peer/customer ASes.

In Tier-2 and Tier-3 cases shown in Figures 2 (b) and (c), AS3307 and AS3557 were first selected as the detection tower, respectively. They share two commonalities. First, the selected ASes will receive the provider route from the destination AS. Second, the selected ASes are either the Tier-2 ASes, or poorly connected to (with one or two connections) Tier-2 ASes.These commonalities are also observed on other detection towers selected by our algorithm.

Above examples show that the more likely a provider is polluted by hijacks, the more likely it can detect hijacks. Figure 3 shows the distribution of likelihood of being polluted for the provider groups in different tiers. Assuming that there is an equal probability for where the hijacker may be on the Internet, we hence define the *likelihood* of pollution associated with a target AS d as the average of the portions of unaffected-source-ASes for all possible hijacker locations. It shows that that highest detection coverage is achieved by Tier-1 ASes since they are most likely to be polluted (our observation (iii)), which is consistent with the observation in [28]. Furthermore, the figure shows that Tier-2s are least likely to be polluted, which has not been reported in the literature to the best of our knowledge, thus have the lowest detection coverage.

Which ASes are selected next after the first detection tower is selected? Figure 1 shows that the second tower selected offers good improvement of detection coverage than towers selected later, especially for Tier-1 and Tier-2 cases. We now investigate the similarity between the towers selected first and second by our algorithm. We define the term *Tier-2 peering set* of the tower, given the key role of Tier-2 ASes in detection coverage. If a Tier-2 AS is selected, then the Tier-2 peering set is the set of the ASes peering with this Tier-2 AS. Otherwise, the Tier-2 peering set is the set of ASes peering with the AS' Tier-2 provider(s)[3]. We compute the *Jaccard coefficient*[4] of Tier-2 peering sets of the first two selected towers to investigate their similarity. The Jaccard coefficient for the first two selected towers on average is 0.18, with maximum 0.27; while the overall Jaccard coefficient for any two Tier-2 ASes on average is 0.46. This result indicates that the first two selected towers have significant different peering sets. In other words, they are diverse from each other.

Why does the coverage gain of using additional towers become marginal after a few detection towers are selected? We noted that the coverage become stable after selecting first few detection towers. The reason is that some hijacking cases are difficult to detect, making it difficult to achieve 100% overall detection coverage, thus there is not much room for coverage increase from the already-high coverage provided by the first few selected towers. We investigated those hard-to-cover hijacking cases, and found that, generally speaking, these are *locally polluted cases*, where only several stub nodes are polluted by the hijackings. Figure 4 shows a real example of locally polluted case. Hijacker AS15227, which has only one provider AS7018, advertises the prefix p belonging to the target stub AS d. AS7018 then has two equally good routes, both from customers and with the same path length of 1. Therefore, AS7018 has a 50% chance to select either path. In case it sticks to the original path learned from d, it will not propagate the fake announcement to other ASes. Therefore,

[3]This definition does not apply on Tier-1 AS since no Tier-1 AS was selected by our algorithm.

[4]The Jaccard coefficient measures similarity between sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets.

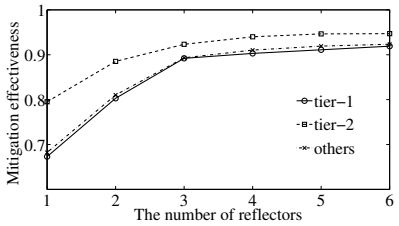

Fig. 5. single-provider, reflector case.

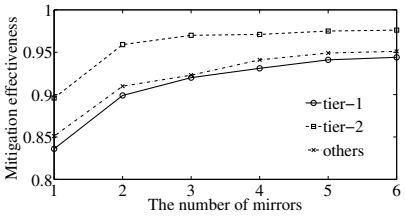

Fig. 6. single-provider, mirror case.

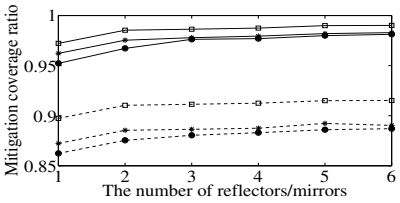

Fig. 7. Multi-provider, reflectors (the three dash lines above) and mirrors (the three solid lines below). Tier-1: Circle, Tier-2: Square, Others: Star.

only AS15227's direct or indirect customer ASes (the four gray nodes in the figure) are impacted in this case, and unless we have detection tower in these ASes, this hijacking will not be not detected.

2) Multi-Provider Stubs: The result of multi-provider stubs are similar to that of single-provider ones. Due to space limitation, we only emphasize two different points here. The first point is that the more providers a stub customer has, the more detection towers are needed. It is because the more providers a stub customer has, the smaller impact a hijacker has, and hence the harder to detect with a small number of detection towers. The second point is how a multi-provider stub d selects the TOWERDEFENSE provider. The answer is that d can choose any of its providers. The detection coverage of using different providers for d are very similar because the detection towers are selected based on the same set of information (e.g., AS topology). Detailed analysis of multi-provider stubs can be found in [30]. In this paper, we are primarily thinking from providers aspect, that is, how to provide such kind of service. From customers perspective, there are other considerations beyond detection coverage (e.g. how much should be paid to enjoy the service) when choosing different providers. It is out of the scope of our paper.

3) Detection Tower Selection Strategies: Based on our analysis results, we summarize the strategies on selecting detection towers for a given service provider X and a given target d. These guidelines help service providers not only understand the usefulness of existing vantage points, but also determine adding new vantage points. But it is just a rough suggestion. More reliable way to selection towers is to run the selection algorithm we proposed. When the service provider has no "complete" AS topology or simply do not want to run our selection algorithm, it can still choose the vantage points based on local topology information of the candidate vantage points according to the following strategies.

1) Select v that has multiple providers and is connected to many peers such that v uses a provider or a peer route to reach as many targets as possible, making it easier to be polluted by the fake routes from peers or customers, respectively. Some (not all) well-connected tier-2 nodes satisfy this requirement.
2) Select v which is relatively far away from d so that AS path to d is more likely to be polluted by a shorter fake route.
3) Select the immediate poorly connected (e.g. single-provider) customer of v as the alternative.
4) Select v that is diverse from existing detection towers. For example, one should avoid selecting v in an AS which is directly connected to an already selected detection towers.

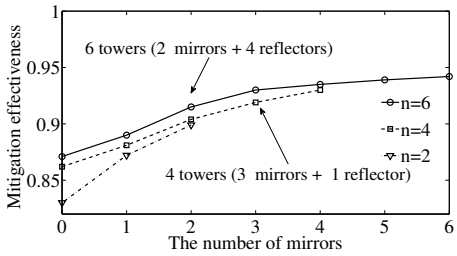

Fig. 8. Combining mirrors and reflectors.

B. Mitigation Effectiveness

1) Analysis Results: Figures 5 and 6 show average mitigation coverage for single-provider stubs and Figure 7 shows those for multi-provider stubs. As expected (recall the study of Figure 3), for both reflectors and mirrors, stub customers of Tier-2 ASes can be better mitigated (e.g. up to 98.2% in Figure 7) than stub customers of other tier ASes with the same number of mitigation towers. Mirror mitigation is always better than reflector mitigation because a successful mirror does not require the path from itself to the target d not to be polluted by hijacking events on d, but a successful reflector does.

To further illustrate the mitigation coverage difference between mirrors and reflectors, Figure 8 shows the mitigation coverage for single-provider stubs which are customers of Tier-1 ISPs when using $n(n = 2, 4, 6)$ mitigation points consisting of m mirrors $(m = 0, 1, ...n)$ and $n - m$ reflectors. We find that for all cases, the mitigation coverage increases as the number of mirrors increases. In addition, the curves are close to each other when the same number of mirrors are used. This observation seems to suggest that the dominant mitigation coverage are contributed by mirrors in these mixed compositions. In other words adding reflectors to a mirror mitigation system has limited marginal benefit.

Figure 9 illustrates three cases for using reflectors in mitigating single-provider stub d connected to ASes of different

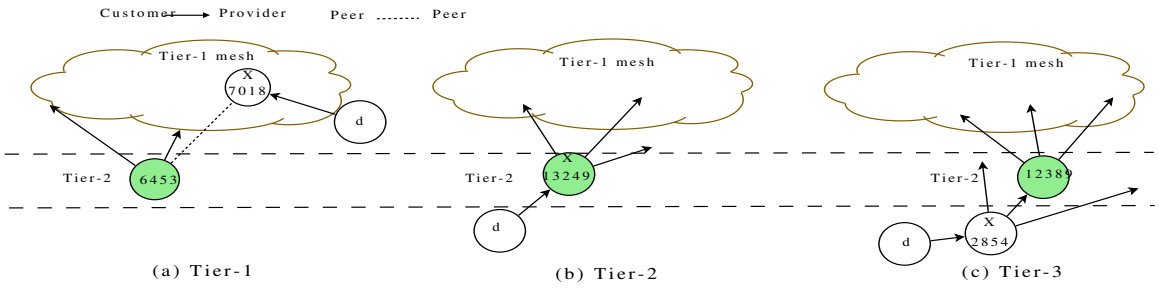

Fig. 9. The examples of reflector selection

tiers. The top choice reflectors are the lightly shaded ASes. The most noticeable commonality among the reflectors is that they are all Tier-2 ASes with many Tier-1 and Tier-2 neighbors. This is also common to all top choice mirror locations as well. The other two commonalities among these reflectors are that (i) they are relatively close (e.g., one or two hops away) to the target d, and (ii) the path between the reflector and d contains, in decreasing preference order, provider, peer, and customer links.

2) Mitigation Tower Selection Strategies: Based on our results, we suggest two general strategies for selecting reflectors and mirrors.

1) Find the reflector r which has smallest chance to be polluted by a hijacking event on the target stub customer d. This is complimentary to the detection selection. It is preferable to select a reflector r (i) of which the origin route from d to r is a customer route than a peer route than a provider route; (ii) which is close to d; (iii) which covers as few number of potential hijacker routes learned from providers and peers as possible. Note that this strategy applies only to reflector selection and is not needed for mirror selection.
2) Find the reflector r which will not be easily hijacked. That is, one need to select a r which reaches as many Tier-1 ASes and other large ISPs as possible via customer routes. In addition, the route from r to each of these Tier-1 ASes and large ISPs should be short. This strategy applies to both reflector and mirror selection.

C. Impact of Incomplete AS topology

It is well known that the AS topology is incomplete [32]. We now evaluate the robustness of our tower selection algorithms. The challenge is that there is no ground true of AS topology available. We have no idea what kind of links are missing. Hopefully, According to the study [32], many peer links between lower tiers' ASes can be missing in the inferred AS topology based on public BGP data. Therefore, We assume that there are $x\%$ of peer links between Others ASes are missing and the missing peer links are randomly distributed. To reconstruct the "complete" AS topology, we randomly select $n/(1-x\%)$ pairs of Others ASes, the two ASes in each of which are not neighbors, where n is the number of inferred peer links in the AS topology. We then add a peer link between ASes in each selected AS pair to the AS topology. We select the towers based on incomplete topology and evaluate the accuracy by simulating the hijack events based on the "complete" topology. Table I shows the average detection and mitigation coverage, when the number of towers are fixed as 6. We observe that the coverage decease when increasing x. It is because the larger x is, the larger the differences are between the topology used to select the towers and the complete topology. We also find that even missing half of peering links, the algorithm has relative high (more than 86%) coverage, indicating that our algorithm is robust to the missing links.

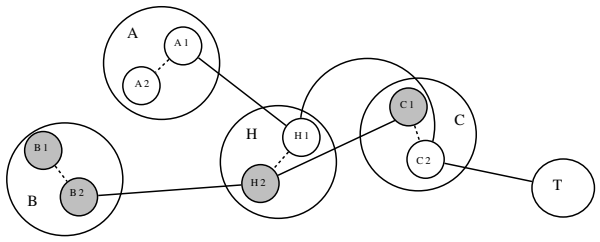

Fig. 10. The examples of route diversity.

D. Impact of Route Diversity

In our evaluation, we mainly assume that (1) the hijacker will pollute all its neighbors to maximize the impact. (2) when one router in the AS is polluted, then all routers in this AS will be polluted. In reality, the hijacker may select some of neighbors to propagate the fake AS path announcement. Moreover, it is possible that some of routers in the AS will be polluted, especially when the AS is large e.g. tier-1 or tier-2 ASes. As a result, different routers within one AS may have different views of routes. Figure 10 shows an example. Assume that AS T is the owner of prefix p. Hijacker AS H announces itself as the prefix owner, and propagate the announcement through edge router $H2$. Routers $B1$, $B2$ and $C1$ are polluted. $C2$ is not polluted, because C is one hop way from both H and T, $C1$ will prefer the routes learned from e-BGP session of T. As a result, AS A is not polluted, AS B is fully polluted, and AS C is partially polluted.

In order to evaluate the impact of these *route diversity* cases on our tower selection algorithm, we conduct the following simulation. We split hijacker AS and each tier-1/tier-2 ASes into two sub-ASes (like in Figure 10). These two parts have equal number of neighbor ASes. We "maximize" the diversity in this way. Therefore, our evaluation in this part already

TABLE I
ROBUSTNESS OF TOWER SELECTION, FACING INCOMPLETE TOPOLOGY

x	0	10	20	30	40	50
Detection tower	.902	.895	.891	.883	.875	.867
Mirror	.953	.950	.941	.933	.929	.920
Reflector	.923	.918	.914	.908	.903	.891

TABLE II
ROBUSTNESS OF TOWER SELECTION, FACING ROUTE DIVERSITY

y	0.1		0.9		1.0	
	APX	OPT	APX	OPT	APX	OPT
Detection tower	.822	.853	.848	.863	.902	.902
Mirror	.906	.927	.932	.944	.953	.953
Reflector	.882	.902	.903	.914	.923	.923

overestimated the impact of AS route diversity. We define the overlap ratio of neighbors as y. Due to the difference of neighbor AS, these two sub ASes may have different view of AS updates. Under this condition, tower selection is more restricted: In order to cover the hijacking events, the detection tower should be in the AS whose both sub-ASes are polluted, e.g. AS B in Figure 10. In terms of mitigation, we assume that mitigation towers should be in the AS whose neither of two sub-ASes are polluted, e.g. AS A in Figure 10.

To evaluate the impact of partial propagation, we compare the detection/mitigation coverage of the towers selected by original simulation environment (APX) and the new one (OPT), under the new and more "real" propagation cases. We fix the number of towers as 6 and tune the parameter y. Table II shows the results. The small value of y means the higher diversity of route views, which means that selection of detection tower and mitigation tower are more restricted, making it harder to select the towers. We find that the smaller y is, the the smaller the detection/mitigation coverage is. We also find that the coverage of APX is slightly lower than OPT when $y = 0.1$ and $y = 0.9$. When $y = 1.0$, the coverages are the same because two sub-ASes have identical view. Given that we have no idea that the real partial propagation looks like, we will still use original methodology in practice.

E. Case Study: How Large ISPs May Improve Protection Effectiveness

We now use a case study to illustrate the value that the TOWERDEFENSE system may offer to large ISPs.

A large ISP often has multiple ASes. Thus it is tempting for such an ISP to simply deploy detection and mitigation points at its own ASes for protecting the ISP's customers. Such a deployment strategy may also seem effective because such ISPs networks often span across large geographic areas or even multiple continents. Our case study is about a large Tier-1 ISP. Despite the fact that this ISP has 20 ASes of its own, Figure 11 shows that the detection and mitigation coverage (averaging over all of its direct stub customers) are very low when only the ISP's own 20 ASes are used, with no additional towers(i.e., 0 on X-axis).

We first investigate how our tower selection algorithms can help improve this Tier-1 ISP's deployment strategy. First, when we start from scratch, 3 ASes are enough to achieve the same coverage as using all 20 existing ASes can achieve. Second, in addition to using self-owned ASes, external ASes can be identified to help improve protection quality quickly. Figure 11 shows how protection quality significantly increases as the number of external ASes are used for deploying detection and mitigation towers.

Next, we use the same Tier-1 ISP as an example to show that TOWERDEFENSE service can be incrementally deployed. Based on the public topology data, in total this AS has 823 stub customers, including 390 single-provider customers and 433 multi-provider stub customers. Initially we randomly select one customer and we pretend this is the first customer signing up for prefix hijacking protection service. We deploy 6 towers using the methods as described before. Next, we randomly choose another customer and pretend that this is a new customer signing up for the service. It may or may not be necessary to add new tower or towers to maintain the overall protection coverage to be not lower than its current vale. Figure 12 shows how the number of towers increases as more and more customers sign up for the service. The gradual slopes of lines indicate that such service can be incrementally deployed as the number of customers increases. Even when a majority of its customers (800 out of 1266) have signed up one by one for the TOWERDEFENSE service, at most 20 towers (9 for detection, 11 for either mirror or reflector) are needed.

V. INTERNET EXPERIMENTS

We evaluate TOWERDEFENSE performance by constructing synthetic hijacking attacks using Internet measurements on Planetlab [33].

A. Experimental Methodology

We conduct our experiments in the following steps. First, we identify a set of target prefixes used in the experiments (343 in total). Then, we select candidate Planetlab nodes (73 in total) to serve as the base of our experimental infrastructure. Each node can serve as detection tower, mitigation tower, traffic source, or hijacker in various attack scenarios. Next, for each target prefix, we select detection towers and mitigation towers among candidate Planetlab nodes using TOWERDEFENSE methodology. As a comparison, we also implemented monitor selection schemes studied in [34]: (1) **random based**: monitor nodes are selected randomly and (2) **greedy link based**: at any time, the next tower is selected with the largest number of unobserved links, given the set of already selected towers. Note that greedy link algorithm can only be used for detection. Finally, using methodology similar to that in [19], we construct all possible attack scenarios among candidate Planetlab nodes and evaluate the performance of TOWERDEFENSE. More detailed experimental settings can be found in [30].

B. Detection Tower Selection Effectiveness

We use the detection method proposed in [18], which uses hop count and path divergence information obtained

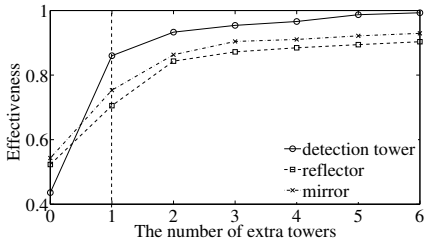

Fig. 11. Coverage vs. # of extra towers.

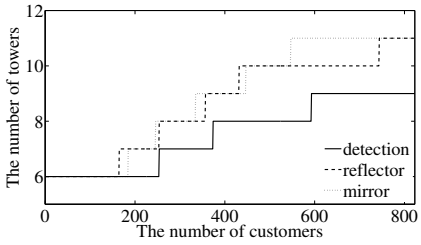

Fig. 12. The incremental deployment.

TABLE III
COVERAGE OF TOWERDEFENSE OVER PROTECTED TARGETS.

	TowerDefense		random		Greedy-link	
	AVG	STD	AVG	STD	AVG	STD
Detection Coverage	.943	.013	.632	.104	.842	.062
Reflector Miti. Coverage	.816	.023	.432	.203	NA	NA
Mirror Miti. Coverage	.846	.022	.443	.228	NA	NA

from the data plane. In addition, we use a fixed number of detection towers (i.e., 6) in the Planetlab experiments because the gains of additional towers become marginal, similar to the simulation results in previous section. Then the average detection coverage for each target prefix is computed.

Table III compares the coverage (average and standard deviation) of detection tower selection using TOWERDEFENSE algorithm, random and greedy-link based algorithm [34]. We observe that our algorithm yields the highest detection coverage. Greedy-link algorithm is better than random algorithm because it tends to maximize the visibility of AS topology. But it is not as good as our algorithm because its optimization goal is to maximize link visibility, rather than hijacking probability of protected targets.

Though we use the detection method proposed in [18] in our experiments, TOWERDEFENSE can adopt any of the existing detection methods [2], [13]–[18], [29]. The only exception is iSpy [20]. iSpy is a data plane prefix hijacking detection method that is designed to be used by the target prefix itself. Another important difference between TOWERDEFENSE and iSpy is that TOWERDEFENSE carefully chooses a small number of detection towers and probes from the detection towers to the target prefix, while iSpy probes from the target prefix to every transit AS on the Internet. Figure 13 compares the coverage of TOWERDEFENSE and iSpy with varying probing costs under default settings. We observe that when the number of probe paths is small, TOWERDEFENSE can achieve much higher detection ratio (the percentage of detected hijacking events) than iSpy. For example, TOWERDEFENSE can achieve over 90% detection ratio by using 5 detection towers, while iSpy can achieve about 50% detection ratio if 5 random transit ASes are probed.On the other hand, we also observe that iSpy can achieve 99.54% detection ratio when all (thousands of) transit ASes are probed. This implies that TOWERDEFENSE is much more cost effective than iSpy, though both methods can achieve comparable detection ratio when probing cost is not a concern.

C. Mitigation Tower Selection Effectiveness

Mitigation Coverage. For each target prefix, we select a fixed number of mitigation towers (i.e., 6) among candidate Planetlab nodes and compute the mitigation coverage for each possible attack scenario. Table III also compares average mitigation coverage achieved by both mirrors and reflectors selected using TOWERDEFENSE algorithm with random algorithm. The result of greedy link is not available because it is designed for detection only. Our algorithm is the best. More specifically, we observe that the average mitigation coverage is about 80% with 6 carefully selected reflectors.

Hijacking Impact Reduction. We measure the impact of a hijacking event by the percentage of ASes from which the path to the target prefix is polluted by the hijacker. We compare the impact of a hijacking event before and after using mitigation towers. Figure 14 shows the hijacking impact reduction when 6 mitigation towers are used in TOWERDEFENSE. We observe that the use of reflectors or mirrors significantly reduced the impact of hijacking events (e.g., from 65% $\sim$ 90% to 10% $\sim$ 15%). Again, the reduction of hijacking impact by using mirrors is more significant than that of using reflectors.

Changes in AS Path Lengths. In TOWERDEFENSE, the impacted traffic is re-routed to or through mitigation towers. We compare the AS path lengths of the impacted traffic before and after using mitigation towers for each target prefix. Figure 15 shows that the average AS path lengths increases 1.7 AS hops and 0.6 AS hops when reflectors or mirrors are used, respectively. Note that a negative value means a decrease in AS path lengths. This is observed for some target prefixes when some mirrors are placed in the upstream providers of the target prefix.

VI. RELATED WORK

A number of solutions have been proposed to proactively defend against prefix hijacking [2]–[12], but the placement and deployment problems are not the focuses of these work. They also need to change router software, router configurations, network operations, or introduce public key infrastructures, and most of them also need explicit collaboration with others, which make immediate deployment very difficult. For example, in the mitigation approach in [12], victim AS needs to collaborate with its previous-arranged "Lifesaver" ASes to remove the bogus route and promote the genuine route.

The hijacking *detection* approaches [13]–[18], [20], [29] use control-plane and/or data-plane vantage points to detect

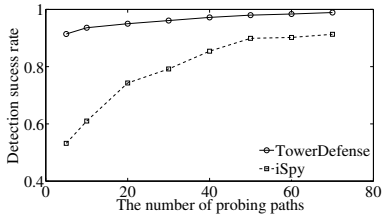

Fig. 13. The coverage of detection tower selection, compared with iSpy

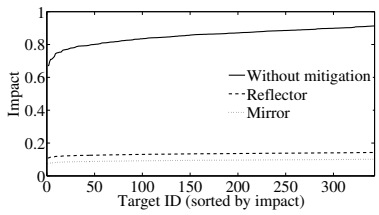

Fig. 14. Hijacking impact reduction using mitigation towers.

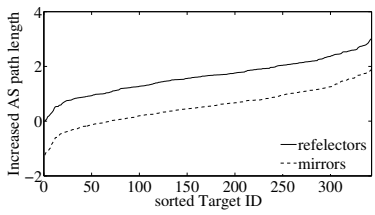

Fig. 15. The change of AS path lengths using mitigation towers

hijacking. However, most of them depends on existing routing information tapping points (e.g. Route Views [25] and RIPE [26] or regulated traffic access(e.g. PlanetLab [33]), which are often not optimum for hijacking detection.

VII. Conclusion

In this paper, we propose the practical deployment strategies for battling against IP prefix hijacking, which we call TowerDefense . We advocate that the best way to move forward prefix hijacking protection is to offer such a protection as a new type of service by existing service providers, and propose a simple heuristic for the placing detection and mitigation agents. Through extensive simulations and large scale experiments, we show that with a small number of detection and mitigation agents deployed at locations selected by our selection algorithms, TowerDefense can achieve high detection and mitigation success ratios. Our case study of one Tier-1 ISP as TowerDefense provider also shows that high success ratios can also be achieved when detection and mitigation points are incrementally deployed.

Acknowledgments

This work was supported in part by NSF grants CNS-0905169 and CNS-0910592, funded under the American Recovery and Reinvestment Act of 2009 (Public Law 111-5), and NSF grants CNS-0716423. We are grateful to Dr. Mickael Meulle and the anonymous reviewers for their comments and suggestions.

References

[1] Y. Rekhter, T. Li, and S. Hares, "Border Gateway Protocol 4," Internet Engineering Task Force, RFC 4271, Jan. 2006.
[2] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. H. Katz, "Listen and Whisper: Security Mechanisms for BGP," in *Proc. USENIX NSDI*, Mar. 2004.
[3] W. Aiello, J. Ioannidis, and P. McDaniel, "Origin Authentication in Interdomain Routing," in *Proc. of ACM CCS*, Oct. 2003.
[4] Y.-C. Hu, A. Perrig, and M. Sirbu, "SPV: Secure Path Vector Routing for Securing BGP," in *Proc. ACM SIGCOMM*, Aug. 2004.
[5] S. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (S-BGP)," *IEEE JSAC Special Issue on Network Security*, Apr. 2000.
[6] K. Butler, P. McDaniel, and W. Aiello, "Optimizing BGP Security by Exploiting Path Stability," in *Proc. ACM CCS*, Nov. 2006.
[7] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin, "Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing," in *Proc. NDSS*, Feb. 2003.
[8] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, A. Mankin, S. Wu, and L. Zhang, "Protecting BGP Routes to Top Level DNS Servers," in *Proc. IEEE ICDCS*, 2003.
[9] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. Wu, and L. Zhang, "Dection of Invalid Routing Announcement in the Internet," in *Proc. IEEE/IFIP DSN*, June 2002.
[10] J. Karlin, S. Forrest, and J. Rexford, "Pretty Good BGP: Protecting BGP by Cautiously Selecting Routes," in *Proc. IEEE ICNP*, Nov. 2006.
[11] S. Y. Qiu, F. Monrose, A. Terzis, and P. D. McDaniel, "Efficient Techniques for Detecting False Origin Advertisements in Inter-domain Routing," in *Proc. IEEE NPsec*, Nov. 2006.
[12] Z. Zhang, Y. Zhang, Y. C. Hu, and Z. M. Mao, "Practical Defenses Against BGP Prefix Hijacking," in *Proc. ACM CoNext*, Dec. 2007.
[13] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, "Topology-based Detection of Anomalous BGP Messages," in *Proc. RAID*, Sept. 2003.
[14] "RIPE myASn System," http://www.ris.ripe.net/myasn.html.
[15] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang, "PHAS: A Prefix Hijack Alert System," in *Proc. USENIX Security Symposium*, Aug. 2006.
[16] X. Hu and Z. M. Mao, "Accurate Real-time Identification of IP Prefix Hijacking," in *Proc. IEEE Security and Privacy*, May 2007.
[17] G. Siganos and M. Faloutsos, "Neighborhood Watch for Internet Routing: Can We Improve the Robustness of Internet Routing Today?" in *Proc. IEEE INFOCOM*, May 2007.
[18] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis, "A Light-Weight Distributed Scheme for Detecting IP Prefix Hijacks in Real-Time," in *Proc. ACM SIGCOMM*, Aug. 2007.
[19] T. Qiu, L. Ji, D. Pei, J. Wang, J. Xu, and H. Ballani, "Locating Prefix Hijackers using LOCK," in *Proc. USENIX Security Symposium*, Aug. 2009.
[20] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush, "Ispy: Detecting IP Prefix Hijacking on My Own," in *Proc. ACM SIGCOMM*, Aug. 2008.
[21] Y. Zhang, Z. Zhang, Z. M. Mao, and Y. C. Hu, "HC-BGP: A Light-weight and Flexible Scheme for Securing Prefix Ownership," in *Proc. DSN-DCCS*, 2009.
[22] "Tower defense," http://en.wikipedia.org/wiki/Tower_defense.
[23] V. Cardellini, M. Colajanni, and P. S. Yu, "Redirection algorithms for load sharing in distributed web-server systems," in *Proc. IEEE ICDCS*, May. 1999.
[24] A. Shaikh, R. Tewari, and M. Agrawal, "On the Effectiveness of DNS-based Server Selection," in *Proc. IEEE INFOCOM*, Apr. 2001.
[25] "University of Oregon Route Views Archive Project," http://www.routeview.org.
[26] "RIPE RIS Raw Data," http://www.ripe.net/projects/ris/rawdata.html.
[27] L. Gao, "On Inferring Autonomous System Relationships in the Internet," *IEEE/ACM Transactions on Networking*, 2001.
[28] M. Lad, R. Oliveira, B. Zhang, and L. Zhang, "Understanding Resiliency of Internet Topology Against Prefix Hijack Attacks," in *Proc. IEEE/IFIP DSN*, June 2007.
[29] H. Ballani, P. Francis, and X. Zhang, "A Study of Prefix Hijacking and Interception in the Internet," in *Proc. ACM SIGCOMM*, Aug. 2007.
[30] "TowerDefense: Deployment Strategies for Battling against IP Prefix Hijacking. (Available from TPC chair)," Tech. Rep., 2010.
[31] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz, "Characterizing the Internet Hierarchy from Multiple Vantage Points," in *Proc. IEEE INFOCOM*, Apr. 2002.
[32] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang, "The (in)Completeness of the Observed Internet AS-level Structuree," in *IEEE/ACM Trans. Networking*, 2010.
[33] "PlanetLab," http://www.planet-lab.org.
[34] Y. Zhang, Z. Zhang, Z. M. Mao, Y. C. Hu, , and B. Maggs, "On the Impact of Route Monitor Selection," in *Proc. ACM IMC*, 2007.