Collaborative Framework for Protection Against Attacks Targeting BGP and Edge Networks

Rahul Hiran, Niklas Carlsson*, Nahid Shahmehri

Linköping University, Linköping SE-58183, Sweden

Abstract

This paper presents the design and data-driven overhead analysis of PrefiSec, a distributed framework that helps collaborating organizations to effectively maintain and share network information in the fight against miscreants. PrefiSec is a novel distributed IP-prefix-based solution, which maintains information about the activities associated with IP prefixes (blocks of IP addresses) and autonomous systems (AS) and enables efficient sharing of this information between participants. Within PrefiSec, we design and evaluate simple and scalable mechanisms that help to protect against prefix/subprefix attacks and interception attacks, and enable sharing of prefix related information related to a wide range of edge-based attacks, such as spamming and scanning. We also include an evaluation of which ASes need to collaborate, to what extent the size and locality of ASes matter, and how many ASes are needed to achieve good efficiency in detecting anomalous route announcements. Public wide-area BGP-announcements, traceroutes, and simulations are used to estimate the overhead, scalability, and alert rates. Our results show that PrefiSec helps improve system security, and can scale to large systems.

Keywords: Collaboration, Information sharing, Interdomain routing, BGP, Prefix hijack, Interception attacks

1. Introduction

Today, organizations and network owners must protect themselves against a wide range of Internet-based attacks. The Border Gateway Protocol (BGP) is susceptible to prefix hijacks, sub-prefix hijacks, and interception attacks [1, 2]. Edge networks and the machines within these networks may be scanned, probed, or spammed with unwanted traffic/mail [3, 4, 5]. In addition, network

☆A preliminary version appeared in the Workshop on Information Sharing and Collaborative Security (WISCS) at ACM Conference on Comp. and Comm. Security (CCS) 2014.

*Corresponding author

Email addresses: rahul.hiran@liu.se (Rahul Hiran), niklas.carlsson@liu.se (Niklas Carlsson), nahid.shahmehri@liu.se (Nahid Shahmehri)

owners must be aware that machines within their networks may be compromised, participate in botnet activities, DDoS attacks, or in other ways cause harm.

Unfortunately, miscreants are becoming increasingly sophisticated and security attacks are no longer isolated events. Instead, attacks often cover multiple domains with differing behaviors in different domains, making it difficult for a single network entity to detect them. Collaboration among network entities provides richer information, and can help detect and prevent such attacks [6, 5]. With an expected increase of cyber attacks and an urgent need for strengthened network security [7], it is important to design systems that help responsible organizations collaborate in the battle against miscreants.

While collaboration among organizations has been proposed, and the value of such collaboration demonstrated (e.g., [6, 8, 9]), it remains an open problem to design distributed mechanisms that provide effective decentralized information sharing among disparate organizations and Autonomous Systems (AS). In this paper, we present the design and data-driven overhead analysis of PrefiSec, a distributed system framework that (i) provides scalable and effective sharing of network information, (ii) provides notification alerts and aggregated evidence information about a wide range of attacks, and (iii) helps responsible organizations to keep their network footprints clean.

Scalable overlay design (Section 3): At the center of our design is a distributed reporting and information monitoring system that allows participating members to effectively share route/prefix information and observations, report suspicious activities, and retrieve information about organizations, networks, their IP prefixes (blocks of IP addresses), and the activities within each prefix. To capture the intricate relationship structure between ASes and their prefixes, as well as the hierarchical nature of the IP space, we design an overlay consisting of complementary Distributed Hash Table (DHT) structures, and a novel distributed Chord [10] extension that provides functionalities such as longest-prefix matching, used in Internet routing.

Distributed alert mechanisms for prefix and subprefix hijacks (Section 4): BGP uses prefix announcements to determine the routing paths that will be taken by Internet Protocol (IP) packets. A (sub)prefix hijack involves an AS announcing a (sub)prefix allocated to another AS without permission. Building on our longest-prefix capable overlay, we design mechanisms for effective and distributed prefix- and subprefix-hijack attack detection and alert notification. We provide the same notification accuracy of origin AS changes as existing central systems (e.g., PG-BGP [11] and PHAS [12]), but distribute the processing across all participants and avoid a single (trusted) point of failure, which typically see extremely high processing load [1]. We also present results considering the size and locality aspects of the ASes that collaborate with each other. With the emergence of regional and national information sharing legislations and agreements at the level of the European Union (EU) and the United Stated (US), for example, the importance for systems to be efficient under both locality constrained and global scale is becoming increasingly important. While such legislations/agreements may help push for the deployment of hijack detection mechanisms, the local biases they introduce may also impact different

systems effectiveness. Our results provide insights into the effectiveness of PrefiSec from such locality perspectives.

Collaborative alert mechanisms for interception attacks (Section 5): Hijacked traffic is even more difficult to detect if the intercepted traffic is rerouted to the intended destination. As such interception attacks typically do not disrupt the service and involve many ASes, whose individual decisions can impact the success of the attacks [13], collaboration is important in detecting and defending against these attacks. Leveraging our overlay and the information that it maintains about AS relationships, we design simple policies and mechanisms for collaborative interception detection, which are low in overhead. In this section we also discuss how alert rate and overhead are affected when the locality aspect of the proposed interception detection mechanism is considered.

Aggregated prefix-based monitoring (Section 6): PrefiSec also provides effective mechanisms for monitoring and bookkeeping about a wide range of edge-network-based attacks, including scanning, spamming, DDoS attacks, and botnet activity. Our prefix-based structure effectively aggregates (often sparse) information from many reporters; e.g., about potential non-legit mail servers originating within a prefix. Such information can help responsible organizations keep their network footprint relatively clean from miscreant activity. With malicious hosts increasingly alternating between malicious behaviors [9, 5], a combined per-prefix repository also helps improve early detection rates across services [6].

Data-driven overhead analysis: Throughout the paper we use public wide-area BGP-announcements, traceroutes, and simulations to estimate the overhead, scalability, and alert rates. Our analysis shows that our distributed solution is scalable, comes with low communication overhead, and allows participating organizations to improve their overall security. For example, our case-based study of the China Telecom incident (that occurred on April 8, 2010) shows that the system would have detected all hijacked prefixes, while maintaining relatively low per-node communication overhead and per-node processing and storage requirements; all non-increasing with increased alliance size.

Previous version: This paper is an extended and improved version of our workshop paper [14]. In this revised version, the evaluation focuses on the collaborating ASes that contribute to RouteViews projects information, rather than on the potential exchange between routers. This better matches our system design and provides better understanding of the collaboration between ASes. We have also added new analysis and discussion of the impact of scale and size of the collaborating ASes on the alert rates, as well as of the impact of locality aspects on the proposed hijack detection mechanisms. To provide insight into potential changes in the alert rates of the proposed mechanism, if applied today compared to in 2010, we have also added analysis with more recent data (from Jan. 2016). The paper has also been strengthened with additional and improved descriptions of the system design, mechanisms for an incentive-based hierarchy extension, and an overhead analysis of IPv6.

Outline: To set the context and provide the necessary background, Section 2 presents a brief introduction to routing attacks and describes different

outcomes of successful routing attacks. As outlined in our description of our contributions above, the following sections then describe and evaluate our system design and our system specific mechanisms for different types of attacks. First, Section 3 presents our scalable overlay design and evaluates its overhead. Then, Section 4 presents and evaluates our distributed alert mechanisms for prefix and subprefix hijacks, and Section 5 presents and evaluates our collaborative alert mechanisms for interception attacks. Finally, Section 6 describes how the system can be used for aggregated prefix-based monitoring, before the paper is concluded with a review of related work (Section 7) and conclusions (Section 8).

2. Routing Attacks

Internet packets are highly vulnerable to routing attacks. This is in part due to the complex nature of Internet routing and in part due to the lack of globally deployed security mechanisms. Today, a typical Internet packet traverses many routers operated by different operators and Autonomous Systems (AS), each with its own separate administrative domain and policies. The packet's wide-area (interdomain) route is determined by the Border Gateway Protocol (BGP), the de-facto interdomain routing protocol used over the Internet.

When BGP was originally designed (in the early 1990s) the Internet consisted of a few ASes and there was an unwritten trust between operators, causing security mechanisms such as basic authentication to be omitted from the protocol. Since then the Internet has grown tremendously and today there are on the order of hundred thousand ASes, each with varying degrees of security and trust.

While many routing incidents go undetected, there have recently been serious incidents that have drawn global attention. These include, for example, a small Indonesian ISP temporarily taking Google offline in parts of Asia, Pakistan Telecom temporarily taking YouTube offline for most of the Internet, China Telecom temporarily attracting and re-routing a large fraction of the world's Internet traffic, as well as various examples of highly targeted traffic interception by networks in Iceland and Belarus [15, 13, 16, 17]. Although not all these incidents were intentional (or can be proven intentional), it is important to be able to effectively detect them when they do occur.

A major vulnerability in BGP is its inability to validate the allocation of prefixes to ASes. This makes it difficult to detect when an AS announces one or more prefixes allocated to other network(s). In a *prefix hijack* the attacker announces a prefix (e.g., a.b.c.d/16) that is actually allocated to a different AS. Depending on AS relationships and how the AS-PATH is propagated through the Internet such attack may attract (or hijack) more or less traffic. In a *subprefix hijack*, the attacker announces a subprefix (e.g., a.b.c.d/24) of a larger prefix (e.g., a.b.c.d/16). Due to the longest-prefix matching rule used by the routers, these attacks may be particularly effective in hijacking traffic.

All the attacks mentioned above may lead to several outcomes. For example, in a *blackholing attack* the attacker simply drops the traffic that it attracts.

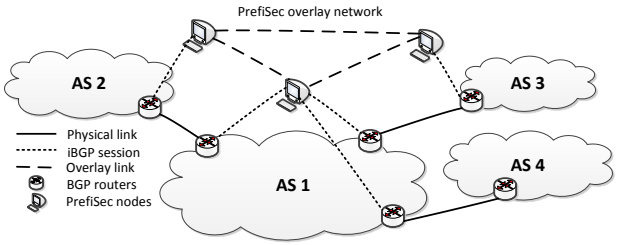


Figure 1: High-level PrefiSec architecture.

In an *imposture attack*, the attacker impersonates the intended destination for the traffic and in an *interception attack* the attacker redirects the traffic to its intended destination, possibly after making a copy or modifying the data, for example. These attacks are particularly stealthy when the users originating the traffic receive uninterrupted service. To help networks protect themselves against these attacks, we present a collaborative, distributed system for information sharing and detection of routing attacks.

3. System Overview

The PrefiSec framework is an application layer service that leverages sharing of network activity observed by routers, network monitors, and other infrastructure. While our design allows both edge networks and ASes to join the alliance, for simplicity of presentation, we assume that a network is an AS with multiple prefixes. Like ASes, edge networks can have multiple prefixes. To map to our AS-focused presentation, edge networks are mapped under a single AS, making them responsible for a fraction of the AS's prefix space. Larger organizations that operate under multiple ASes can simply be considered as multiple members.

Figure 1 provides an overview of the PrefiSec architecture. Here, AS1-AS3 operate separate nodes in the PrefiSec overlay network. We assume that trusted personal relationships among network operators are used to create the overlay network. (A multi-tiered extension is also discussed.) PrefiSec is designed to effectively share and manage any information about ASes and their prefixes. As an example, we present mechanisms and policies designed to effectively detect and/or raise alerts about potential interdomain routing attacks. Relying primarily on reports about origin AS and AS-PATH announcements, we assume that each participating AS collects (e.g., [18, 19]) and share selected BGP updates from its edge routers, for example.

3.1. Distributed overlay

Scalable overlay structures: To keep track of the activity associated with each organization and its IP prefixes, we maintain two complementary distributed structures.

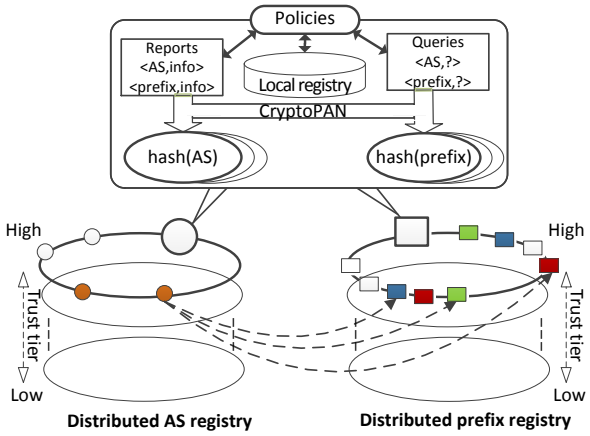


Figure 2: Overview of the framework, its key components, and structure.

- **Prefix registry:** We design a novel Chord-based [10] DHT, which stores prefix origin information (e.g., prefix-to-AS mappings) and observations of edge-network miscreant activities (e.g., scanning, spamming, etc.). The registry keeps track of the prefix hierarchy, and uses a distributed longest prefix matching algorithm for efficient insertion/retrieval.

- **AS registry:** A second Chord-based DHT is used to store information about ASes, their relationships, and AS-to-prefix mappings.

Figure 2 provides an overview of our PrefiSec framework, and shows how the two registry structures are linked by the prefix-to-AS and AS-to-prefix mappings (pointers in figure). Here, a participating member operates a node in the distributed AS registry and one node in the distributed prefix registry (e.g., the large circle and large rectangle, respectively, in the bottom-half of the figure) according to a set of built-in policies and locally stored/retrieved information (as shown in the upper-half of the figure). Reports and queries with shared information and observations are used to populate the registries. Incremental deployment is easily achieved by adding/removing nodes to/from these structures, as members join/leave the alliance.

Distributed information sharing and aggregation: Members share information about prefixes and ASes using *reports* directed to dedicated *holder* nodes (determined based on the reported AS or prefix). For example, suppose a node in PrefiSec receives a BGP update for an AS-prefix pair $< AS1, P1 >$ for the first time. The node will hash the AS number and report the announced prefix to the holder of $AS1$ in the AS registry. Similarly, the node will extract the last IP address from the prefix $P1$ and report the announced origin AS to the holder of $P1$ in the prefix registry. Each holder node is responsible for many ASes and prefixes, and for each AS or prefix, the holders aggregate the information from many reporters. The holder nodes can help the other alliance members (i) by answering direct queries, and (ii) by creating and forwarding ag-

gregated summary reports. Similar to publish-subscribe systems, members can also subscribe to summary reports. We expect that responsible organizations, wanting to keep their network footprint as clean as possible, subscribe to their own prefix and AS information.

3.2. Distributed prefix registry

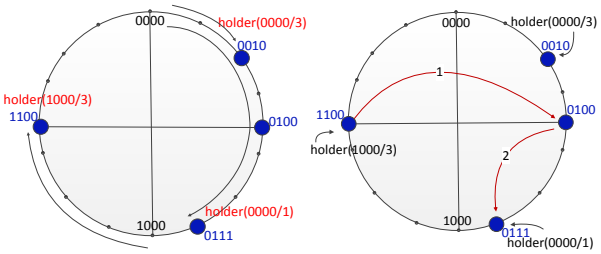
For our AS registry, we use Chord [10] more or less "out of the box". We pick a circular identifier space large enough to uniquely specify any AS (e.g., based on its AS number). For the prefix registry, on the other hand, Chord's (flat) circular identifier space does not naturally capture the hierarchical relationships between prefixes, and must be modified.

Ideally, prefixes of any length should be uniquely assigned to holders, and, given an IP address, the structure should return the holder of the longest-matching prefix. For example, for address 123.123.123.23, prefix 123.123.123.0/24 should be given priority over prefix 123.123.0.0/16. This section describes how we extend Chord to achieve unique and consistent longest-prefix-based assignment and lookup.

Longest-prefix discovery: Global IP-to-prefix lookup queries are resolved using a two-level greedy routing approach. At a high level, we first forward the query to the potential candidate holder h_k of the longest possible prefix of length k, if that prefix exists in the DHT. If h_k is not aware of such a prefix, it forwards the query to the next candidate holder h_{k-1}, which would be responsible for the next longest prefix (of length $k-1$), and so forth, until a prefix is found. For each such high-level forwarding step, multiple regular (low-level) Chord forwardings may be needed. Since /24 typically is the most specific prefix allowed by modern BGP routers, we use $k = 24$ as our initial choice for k.

Holder assignment: Our system defines the holder of a prefix as the node responsible for the last IP address in the prefix. Given a clockwise identifier space, only this choice ensures that the next candidate holder for a prefix of length $k-1$ is ahead of (or the same node as) the holder of the prefix of length k. With this selection of the holder node, in the majority of cases, the next candidate holder for a prefix of length $k-1$ is the same as the holder of the candidate prefix of length k (e.g., in 50% of the cases the last significant bit in the prefix of length k is a 1), and in the other cases, the next node is located in a region of the identity space for which the node has many shortcut pointers.

Example: Figure 3 presents a simple toy scenario, with a total identifier space of $2^4 = 16$ and four nodes: 0010, 0100, 0111, and 1100. Figure 3(a) shows how the prefixes 0000/3, 0000/1, and 1000/3 are assigned to the nodes 0010, 0111, and 1100, respectively. Figure 3(b) shows the high-level messages when node 1100 queries for the longest-prefix match for address 0011. In this case, node 1100 first uses Chord routing to route the query to the node (0100) responsible for the last address (0011) in the prefix 0011/4. When node 0100 receives this query, it observes that it does not have any entries for candidate prefixes 0011/4 and 0010/3, though it would be responsible for both. It then determines that the next biggest range is 0000/2 and uses Chord to route to the last address (0111) in this range. While node 0111 does not have an entry



(a) Holder assignment (b) Query forwarding

Figure 3: Holder assignment, prefix mapping, and longest-prefix query routing.

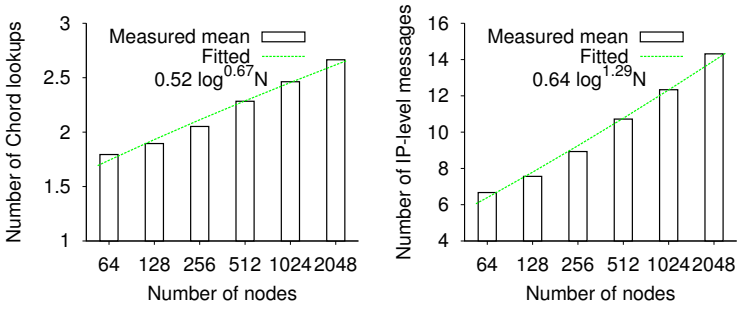
for 0000/2 it is in fact the holder of prefix 0000/1, and can resolve the original query.

Reliability: To ensure efficient recovery at node departures, Chord typically copy the information stored at a node to its successor. For additional reliability, load balancing, and to ensure that no single node is responsible for the entire evaluation of a prefix, multiple holders per prefix are used. Figure 2 shows two holder nodes per AS (e.g., brown circles) and prefix (e.g., red rectangles). Here, CryptoPAN [20] is used to find additional holder nodes for each prefix.

CryptoPAN is a prefix preserving IP address anonymization scheme. With CryptoPAN, any secure stream and block cipher (e.g., an Advanced Encryption Standard (AES) cypher by the National Institute of Standards and Technology (NIST)) can be used to map IP addresses in a one-to-one manner such that two IP addresses that belong to the same $/k$-subnet also are part of the same $/k$-subnet in the new address space. This property allows us to ensure that the hierarchical features of our prefix registry are preserved when applying CryptoPAN to the original IP prefix (or address) in order to obtain H new keys (IP prefix). Using hash-based replication, load balancing is provided complementary. In general, nodes should query multiple holders and inform holders about potential inconsistencies, which may need to be resolved.

Local registry and optimizations: Two optimizations help reduce the Chord-related lookup overhead. First, each node maintains a local registry (Figure 2) with information about the prefixes and ASes that it sees, records statistics for these, and then informs the appropriate holder nodes. The system operates according to a soft-state protocol, with a time-to-live-based cache, and updates entries when changes are detected. A node that has out-of-date information can easily and quickly update its local registry (e.g., prefix tables) using the global DHT registries.

Second, when additional storage overhead is acceptable, existing 1-hop routing optimizations [21] can be used to reduce each lookup to a single hop. While such schemes require each node to have a pointer to every alliance member, Gupta et al. [21] show that the use of slice leaders allows timely, efficient, and scalable updating of the membership pointers and responsibilities under node churn, even for membership sizes up to a few million members. With much



(a) Chord (mean) (b) IP-level (mean)

Figure 4: Number of Chord lookups and IP-level messages to resolve a query.

fewer existing ASes, and on the order of half a million routable prefixes, we foresee these optimizations to be feasible down to the granularity of ASes and the prefixes seen by most core routers.

3.3. Overhead analysis

To evaluate the scalability of the routing overhead associated with longest prefix matching, we use a modified version of PlanetSim [22] to simulate the path the query message takes in alliances with varying sizes. The global repository was populated with all public routable prefixes that were available from the Cyclops project[1] on Sept. 23, 2012, and node identifiers were assigned at random for each simulation.

Figure 4(a) shows the number of Chord lookups for overlays with different numbers of alliance members. For each alliance size, we simulate the query path for one million random IP-address pairs and report the average. We also include a best fit curve of the form $c\log^{\alpha} N$, where α is a scale parameter. Figure 4(b) shows the corresponding statistics for the overall number of IP-level messages (without use of 1-hop optimization). Our fittings suggest that the power α is roughly 0.67 and 1.29, respectively, for the two metrics. The two metrics are identical for the case in which we use 1-hop optimization (equal to values in Figure 4(a)).

Finally, per-node storage overhead scales as $O(PH/N)$, and forwarding tables as $O(\log N)$ or $O(N)$, depending on whether 1-hop optimization is implemented. To put the per-node storage overhead into perspective, consider a scenario in which there are $P = 0.55$M prefixes, $H = 5$ holder nodes, and $N = 100$ alliance members. In this case, each node must on average store 25K prefixes, substantially less than the number of prefixes stored on a typical core router.

[1] Cyclops project, `http://cyclops.cs.ucla.edu/`, Sept. 2012. (This list included roughly 0.55 million prefixes.)

3.4. IPv6 discussion and overhead analysis

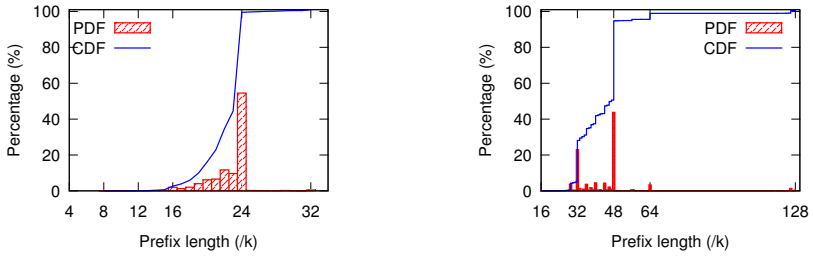
Thus far we have focused on IPv4. While IPv4 still is the dominant IP protocol, the use of IPv6 is gradually increasing [23, 24]. In this section, we discuss how the framework extends to IPv6 and how the corresponding overhead scales in this context.

First, note that IPv6 has a similar hierarchical address space as IPv4. This allows for an easy mapping to the address space and the framework can be implemented using the same general structure and mechanisms. However, given the much larger address space (i.e., 2^{128} compared to 2^{32}), it is important to also take into account how these addresses may impact the system's scaling properties.

Let us therefore consider a worst case analysis of the search time to find the holder node of a prefix. This can be calculated as approximately bounded by $O(\frac{m}{2} \log N)$, where m is the prefix length range for which mapping is required and N is the total number of nodes in the overlay network. This expression is derived based on the observations that there are at most $m/2$ Chord steps, each requiring at most $O(\log N)$ IP-level steps. Here, the division by two is based on the observation that with our assignment of holder nodes and clockwise search, the next holder candidate is located on the same holder candidate for the step with $k' = k$ as for the step with $k' = k - 1$ with probability at least 50%. Also, note that this is a worst case expectation analysis. For example, due to this design choice, all Chord searches after the first would require (much) less than $O(\log N)$ hops. Again, please refer to the previous two sections for details and examples regarding the clockwise identity space, holder assignment, and how these design choices reduce the search space.

Now, given that routers should not forward IPv4 prefixes more specific than /24, for IPv4, it can be argued that k is upper bounded by 24; giving us a max range of $m \leq 24$. Similarly, it has been argued and observed that routers should not globally propagate IPv6 prefixes more specific than /48 [25]. Motivated by this observation, we typically would have $m \leq 48$ for IPv6. This results in roughly a doubling of the m term and the overall overhead expression, given a fixed number of nodes N. However, it is possible that this doubling may be offset by a reduction in the IP fragmentation, partially caused by a lack of IPv4 addresses. Overall, these observations suggest that there may not be any major changes in the number of hops needed when switching to IPv6.

At this point it should be noted that some networks use more specific IPv6 prefixes than /48. Although the original recommendations suggested that end sites would be given their own /48 prefixes [26], the choice of how much address space should be assigned to end sites has later been deferred to the operational community [27]. To better understand what a typical router may see, we performed an analysis of the 36,386 IPv6 prefixes seen by the RouteViews servers on Nov. 3, 2016. Figure 5(b) shows the point distribution function (PDF) and cumulative distribution function (CDF) of the observed prefix lengths, with the x-axis shown on linear scale (but with tics for the most relevant prefix lengths) and the y-values translated to percent. As a reference point, we also include the



(a) IPv4 (b) IPv6

Figure 5: Frequency of IPv4 and IPv6 prefixes.

corresponding distributions for the 660,659 IPv4 prefixes (Figure 5(a)) seen by RouteViews 2 (RIB data) on the same day. For IPv6, the most common prefixes were: /48 prefixes (43.7%), /32 (23.0%), and prefixes with lengths in between these two (22.3%). In total, only 6.2% were more specific than /48, with 3.0% of these being of length /64, and only 2.0% being more specific than /64. The relatively rare occurrences of more specific prefixes suggest that many organizations still follow the old recommendations. Of course, it should be noted that these many smaller prefixes may be hidden since routers are not supposed to forward prefixes more specific than /48. At the other end of the spectrum, we observe only 4.8% prefixes shorter than /32. With only a small number of prefixes outside the /32 to /48 range, we note that many prefixes queries will be resolve in less than 8 (i.e., $m/2 = 16/2 = 8$) Chord steps. While this is roughly twice the most frequent range of IPv4 prefixes (i.e., /16 to /24, as per Figure 5(a)), we note that the actual number of Chord steps likely is even smaller in practice (e.g., due to the sparse allocation of prefixes, and as illustrated by our scaling analysis of IPv4, for example).

In addition to limited changes in the number of Chord steps and good scaling of the IP-level steps, we note that the larger IP addresses may result in somewhat larger packets. However, in general, the overhead associated with carrying the IP addresses and prefixes themselves is negligible compared to the information itself that is shared within the framework. We therefore conclude that PrefiSec would be easy to transition to IPv6 and that any additional overhead would be limited.

3.5. Policies and service implementation

Basic building blocks: As part of providing the high-level services of detecting attacks, the prefix registry and AS registry also implement four effective distributed services that can be used as building blocks for these and other high-level services: (i) IP-to-prefix mapping, (ii) prefix-to-AS mapping, (iii) AS-to-prefix mapping, and (iv) other per-AS and per-prefix information extracted and stored in the repositories. The registries are updated as members observe new mappings, and the holder nodes can easily aggregate sparse information; e.g., to identify and store information about ASes that likely are Internet eXchange Points (IXPs) or siblings.

High-level services: Building on our scalable overlay, we present mechanisms and policies (Figure 2) that allow participating organizations to collaboratively detect and raise alerts about a wide range of attacks. Central routing-related detection mechanisms and policies are built into the overlay itself, whereas high-level mechanisms and policies that help to provide additional services are built on top of the overlay, each leveraging the scalable system design. The system provides scalable detection and alert notification services for three broad classes of attacks: prefix and subprefix hijacks (Section 4), interception attacks (Section 5), and aggregated prefix-based monitoring (Section 6).

Incentive-based hierarchy extension: Additional services are possible to build into the system. For example, our design easily extends to a multi-tiered trust hierarchy (in which nodes are promote/demoted between tiers based on their reporting [28], for example). While such extensions can be important for membership and trust management, for the purpose of our evaluation, we will we assume that all nodes belong to the same tier (setup based on trusted personal relationships, for example), and focus on the scalability and overhead of the system design.

3.6. Membership discussion

This paper focuses on the scale and overhead of our distributed system design. However, also the membership management and active participation of the collaborative parties can play an important role in how effective collaborative systems are in practice. While the details of how to best implement membership management policies are outside the scope of this paper, we include a brief disucssion how our system design and the hierarchical extension above provide system administrators with flexibility to optimize the use of the system for their particular purposes. At one end of the spectrum, relatively centrally controlled policies can be used, in which membership is controlled by one or more organizations that invite other members to the collaboration. At the other end of the spectrum, the system may instead be operated based on a set of open policies determining how to promote/demote participants based on their etiquette, behavior, and/or contributions. Naturally, the first example type may be more desirable when there are legal legislature for who particular information could be shared with in the future, whereas the second example type may be more inclusive, instead focusing on promoting and incentivizing good AS behavior among participants.

Two-tier example: While many implementations and policies are possible, in the following, we briefly describe one candidate policy based on a simple two-tier hierarchy with three classes of members. First, there is a core-group of by-invitation-only members, which are running the system based on mutual trust. These members also participate in the (more trusted) top-tier. Second, there is a group of regular top-tier members that have been promoted (from the lower-tier) to participate in the same top-tier as the core group. These members have been promoted from the lower-tier based on an internal membership evaluation (described next). Finally, there is a group of lower-tier members that yet have to be promoted, or that have been demoted from the top-tier. In general, we expect

the top-tier members to be evaluated primarily by other top-tier members, while the members in the lower-tier may be evaluated by members from both tiers, based on some weighting function.

Membership evaluation: For evaluation of members, a reputation-based model such as that described by Duma [28] can be used. This model uses a dynamic trust metric that is resilient to oscillatory behavior. The model includes a short-term trust factor, a long-term trust factor, and a penalty factor that can be applied either to the short-term or the long-term trust factor. Furthermore, we expect that members are evaluated based on their behavior along many different dimensions, with each coalition being free to set their own weights for the different dimensions. For example, some coalitions may want to penalize members found making routing attacks more than members found harboring a smaller subset of spammers.

Evaluation dimensions: In general, there are four dimension classes along which we expect members to be evaluated. First, as described, the organizations can be evaluated based on others' reports about their prefixes. In addition to creating summary reports about prefixes, the holder nodes can support/suggest potential promotion/demotion cases. Second, organizations are expected to quickly respond to alarms raised by others related to miscreant activity in their network, so as to mitigate the effect that their networks have on others. The holder nodes are again in a great position to evaluate such compliance.

Third, members can be evaluated based on the reports they produce. For example, an organization with a statistically significant number of deviating reports (or prefixes for which it appears to have deviating reports) can be flagged as a deviating reporter. Since such differences may be the effect of a network being more sensitive to attacks, or the network having been the target of more extensive DDoS attacks, for example, these cases typically would require further investigation before any potential promotion/demotion should be considered.

Finally, we expect that some alliances would select to evaluate the holder nodes of prefixes (and ASes) based on their summary reports. Similar to the regular reports, these alliances can leverage that each prefix (and AS) has multiple holder nodes and flag organizations with significantly deviating statistics. While the number of holder nodes is significantly smaller than the number of organizations that typically would evaluate a prefix, we note that the holder nodes (due to the use of cryptoPAN) typically have many prefixes for which they can be evaluated and that the set of holder nodes for each of these prefixes are likely to be different.

Clearly, there are many ways that the above policies can be implemented so to best take into account the above dimensions and the goals of each collaboration coalition (e.g., central vs. decentralized policies). While the structure and information available in our design provide great flexibility and scale, the design and evaluation of individual policies are out of the scope of this paper, and is therefore left as future work. However, it is important to note that the information required to calculate the above dimensions are readily available in our system. For example, the statistics for the first two dimension types are available at the holder nodes themselves. For the third dimension, the holder

nodes (of the reporter to be evaluated) can share statistics through the holder nodes of the reporter, for example. Finally, for the fourth evaluation dimension, the holder nodes (of the holder node to be evaluated) can retrieve and compare summary report from the evaluated holder node and any other holder nodes of the prefixes owned by this holder node. In all cases the use of holder nodes help spread the load and improve the scalability of the system. The use of multiple holder nodes and CryptoPAN also makes the system more difficult to manipulate, regardless of the policy design and the number of tiers, since there are multiple holder nodes (each selected at random with CryptoPAN) for each entity to be evaluated.

Again, for the reminder of the paper we will focus on the single-tier case.

4. Prefix and subprefix hijacks

In contrast to the central processing of prefix origin history used by systems such as PG-BGP [11] and PHAS [12], our system distributes the responsibility and processing of prefixes among holder nodes. These nodes act as information aggregators that maintain history for each prefix, allowing us to improve the scale and accuracy compared to what is possible with central approaches. By distributing the responsibility across multiple holders, PrefiSec also avoids a single point of failure or trust.

4.1. Policy overview

Prefix hijack: We design a distributed prefix hijack detection policy based on PHAS [12] and PG-BGP [11]. These mechanisms keep track of the set of origin ASes for each prefix and raise alerts when changes are detected in the origin set. As discussed in Section 3, in PrefiSec, each participating organization operates one *node* in the AS registry and one node in the prefix registry. The *holder* node of each prefix performs information aggregation and evaluation for that prefix, but is also responsible for detecting when there are changes in the origin AS for a prefix, as well as notifying the previous origin AS of the prefix when a new AS claims ownership of the prefix.

An overview of our hijack alert notification policy is given in Figure 6. The policy is invoked at a node in the alliance network when it sees a new prefix p, a new origin for a prefix p, or when the TTL for the prefix p expires (step 1). The node prepares a query with this information for the holder of prefix p (step 2), and the query is forwarded to the holder of prefix p (step 3) over the overlay network (Section 3.2).

For each prefix p, the holder node tracks the ownership set $A_p(t)$ over some time window of duration T. If the holder sees a change in the origin set, the current owner(s) of the prefix are notified and the ownership set $A_p(t+\epsilon)$ updated (step 4). The case when the prefix has not been previously observed is treated as a case of a potential subprefix hijack and the subprefix hijack policy is invoked at such times.

Subprefix hijack:

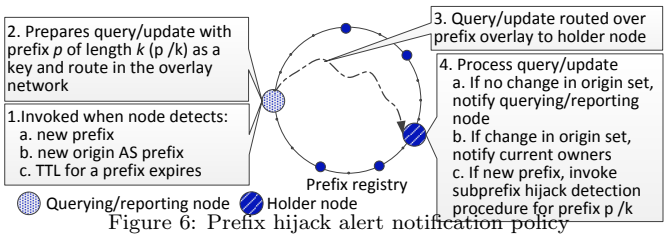


Figure 6: Prefix hijack alert notification policy

When a prefix is observed for the first time, it is important to determine what less specific prefix this may be subprefix hijack attack on. We refer to such a prefix as a superprefix of the newly observed prefix. At the time of such occurrence, our distributed policy finds the immediate superprefix of the announced subprefix and notify the origin AS for the superprefix about the announcement. The origin AS for the superprefix is typically in the best position to determine if the announcement is part of a subprefix hijack attack, or whether the announcement is legitimate and authorized by the origin AS of the superprefix.

Figure 7 provides an overview of our subprefix hijack alert notification policy. The subprefix hijack policy is invoked by a holder node $h_{p'}$ when it receives a prefix query for prefix p' and does not have an entry for this prefix (step 1a). At this time, holder node $h_{p'}$ creates a superprefix query (step 3) and uses Chord (step 4) to send it to the next potential candidate, if needed. To find the next node to forward the query (step 3), the holder $h_{p'}$ reduces the prefix length, say k', of prefix p' by 1. Say prefix p'' is the new prefix with prefix length k'' (step 3.1), the holder node then checks if it is the holder for the new prefix created (step 3.2).

When the query arrives at this holder node, it again invokes the subprefix hijack detection procedure (step 1b). The new holder node checks if it has records for the new prefix (step 2). If the queried holder node has a record for the new prefix, the holder node will send the response and quit the procedure (step 2a). However, if it is not the holder, a new query will be prepared (step 3) that will be routed over the overlay network, with the new prefix p'' as the key (step 4). The process continues recursively until the superprefix p'' for prefix p' is found. When such superprefix is found, the holder node $h_{p''}$ reports owner set $A_{p''}(t-\epsilon)$ for prefix p'' about subprefix p' and the claimed origin set $A_{p'}(t+\epsilon)$.

4.2. Case-based overhead analysis

For our analysis, we examine the announcements seen around the time of the China Telecom incident [13] (April 8, 2010). This day, China Telecom announced origin of approximately 50,000 prefixes originated by others.

Giving consideration to the overhead both when networks are under attack and under normal circumstances, we use the routing tables and updates seen at all six servers participating in the RouteViews project during the first two weeks of April, 2010. We base our original ownership lists on the RIB table

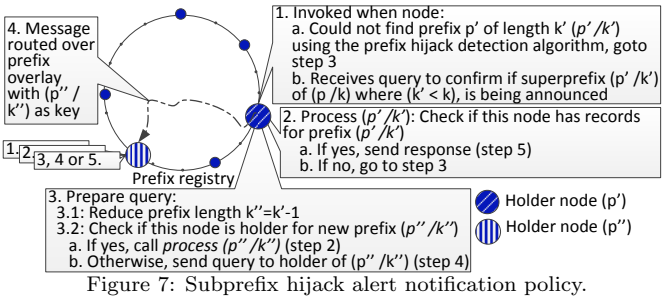
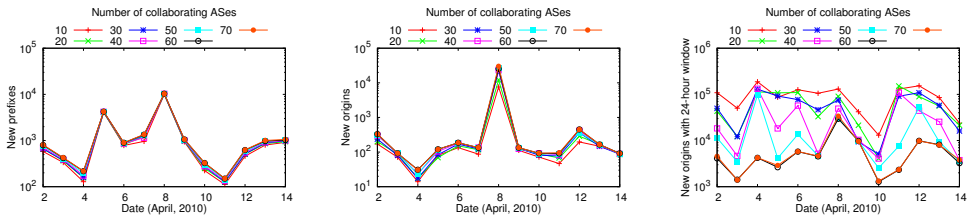


Figure 7: Subprefix hijack alert notification policy.



(a) New Prefixes (b) New Origins (c)Alert rate with-24 hour window

Figure 8: Time-line of anomalous origin reporting.

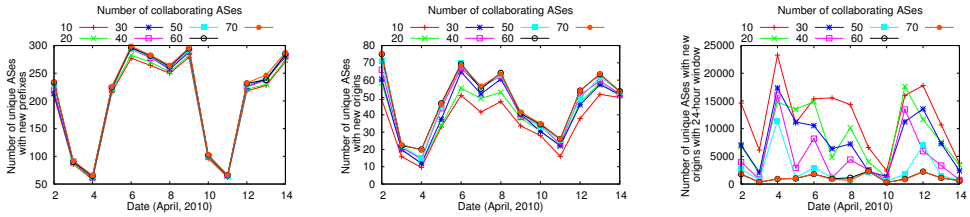
data from April 1, and then use the BGP updates observed during the following period.

Figure 8 shows alert rates starting from April 2, 2010 up to April 14, 2010 (including April 8, 2010, when China Telecom incident occurred). Results are presented for increasingly large alliances of ASes that contribute to the Route-Views project. At the time of the attack, RouteViews operated six servers: RouteViews 2, Linx, Paix, Dixie, RouteViews 4, and Equinix. Together these servers had 100 vantage points that belonged to 73 unique ASes. Of these, 38 are NA-based, 21 EU-based, and 14 map to other geographic regions.

Normal conditions:

The traffic overhead is very small compared to that of PHAS and other techniques that would use central processing. For example, on April 7, 2010, PHAS would have required all 23 million announcements to be forwarded to and processed on a single node (totaling 867MB compressed or 3GB uncompressed data, if using data from all the vantage points. The load scales proportionally with more members. In contrast, with PrefiSec, an alliance of 10 ASes would make 87 prefix queries (due to prefixes with a new origin: "new origin") and 958 subprefix queries (due to new prefixes seen: "new prefixes") to the overlay on April 7, 2010, as seen in Figure 8(a) and 8(b).

Furthermore, with an alliance of 70 ASes, out of all queries generated by individual nodes, 134 and 1,354 queries would eventually result in prefix and subprefix alerts, respectively. The Number of alerts can be further reduced by aggregating messages to the same AS. For example, on April 7, 2010, with 70 ASes collaborating, the alerts concern 56 and 283 unique ASes for prefix and subprefix alerts respectively, as seen in Figure 9(a) and 9(b). For these statistics,



(a) New Prefixes (b) New Origins (c)Alert rate with-24 hour window

Figure 9: Number of ASes affected by anomalous origin reporting .

the superprefix was found using Cyclops data and mapped to an AS using the RIPE *whois* database. These results show that the number of alerts processed by holders scales very nicely with the alliance size. In fact, with the corresponding sub-prefix policy invocations being distributed across holder nodes, we note that the alerts generated per holder node reduce even more (faster than $1/N$). This additional reduction is achieved by distributed information aggregation at holders.

Figure 10 characterizes the overhead of such prefix insertions, as measured by the distance (in prefix lengths) between the two holders for prefix p (to be inserted) and the longest-matching prefix p' for which prefix p is a subprefix. The list of prefixes to be inserted is based on the assumption of collaboration among 33 ASes (37 unique vantage points) that contributed to the *RouteViews 2* server during the China Telecom incident. We use three reference baselines: the RIB of the server itself, the combined RIBs of four different servers (100 vantage points belonging to 62 ASes), and the global Cyclops database. We note that prefix length differences can be substantial, but decrease with larger alliance sizes. Note that similar observations were made when the prefix list to be inserted was created by assuming collaboration between different ASes that contribute to the RouteViews project.

Referring to Figure 8(c), we can also see that the number of updates to the registry if using a (small) 24-hour window is much greater than if also taking into account the RIB information one week earlier (as per the much smaller values for the "New origins" statistics in Figure 8(b)). However, aggregating alerts to the same ASes may lead to significant reduction in the number of messages required to be sent even when a shorter history is used. For example, if aggregating the alerts for April, 7, 2010, the number of alerts within the corresponding 24-hour window can be reduced from 4,615 (Figure 8(c)) to 824 (Figure 9(c)). Of course, using an adaptive window approach may lead to additional improvements [12].

Until now we have focused on the number of collaborating ASes. Figure 11(a) shows the effect of the size of the collaborating ASes themselves on alert rates. We define the size of an AS based on the number of neighbors it has and refer to that number as the degree of that AS. In this experiments, for every degree threshold shown in the label, ten ASes with a degree of at least X are selected at random. Here, the largest displayed threshold is picked so that the selection set includes exactly ten ASes, and the following thresholds

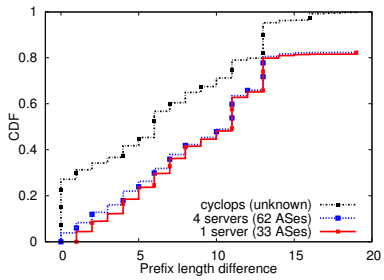
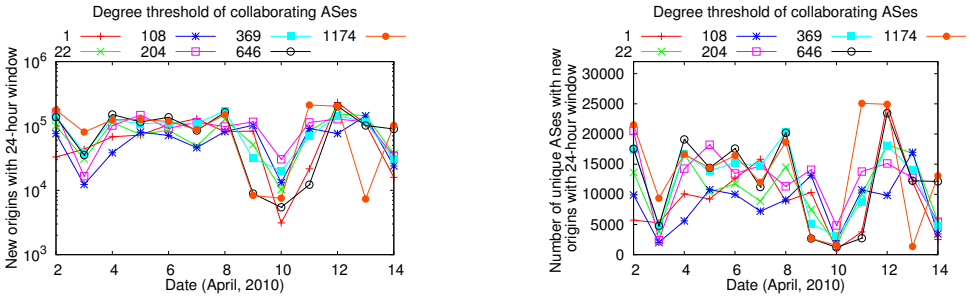


Figure 10: Cumulative Distribution Function (CDF) of the distance to the closest prefix in the global prefix registry, of newly observed prefixes when collaboration between 33 ASes that contribute to the *RouteViews 2* server is considered.



(a) Alert rate with-24 hour window (b) Alerts for number of ASes with 24-hour window

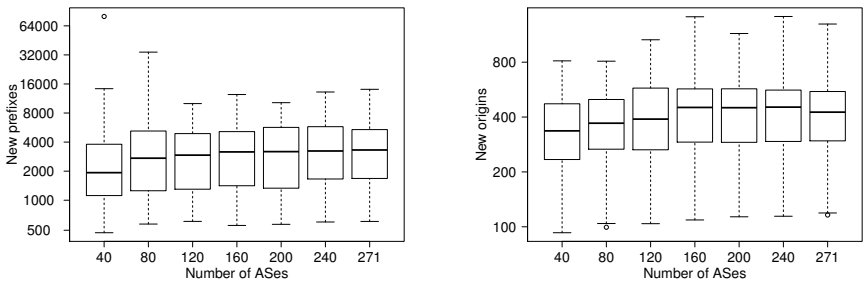
Figure 11: Effect of degree threshold (size) of collaborating ASes on anomalous origin reporting.

(in decreasing size) are picked so as to roughly double the selection set for each point. The degree threshold of one is included as a reference point.

Day of incident: Our overlay allows effective collaborative detection of prefix and subprefix attacks. In fact, during the day of the incident the alliance would raise 40,575 alarms, including alarms for all 39,094 unique prefixes that had the specific signature associated with the incident [13].

Referring back to Figure 8 we note that there is a significant increase in traffic overhead on the day of the incident (April 8, 2010), but that the reporting overhead quickly decreases after the incident. We also note that our system would easily handle such an increase. First, only the prefix holders would need to communicate with the owners of the hijacked prefixes. Second, the holders can easily and quickly sanity check the claims, using the AS registry. China Telecom would have quickly been flagged and additional care could be taken until authenticity had been confirmed or a certain period of time had elapsed. Finally, as seen by the smaller "New prefixes (unique ASes)" (Figure 9(a)) and "New origins (unique ASes)" (Figure 9(b)) statistics, the number of alerts to be sent can be reduced by aggregating messages to the same AS. Similar observations can be made from the Figure 11(b).

Present day (January 2016): Figure 12 presents per-day statistics from



(a) New Prefixes (b) New Origins

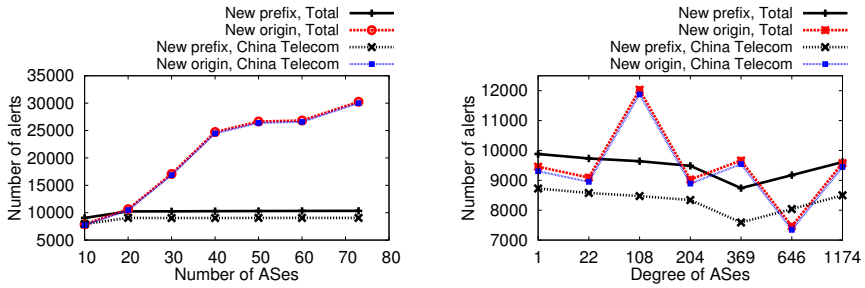
Figure 12: Impact of alliance size.

the eighteen (18) monitors that remained active throughout January 2016, including the subset used for the 2010 data. This dataset consists of 601 vantage points that belong to 271 ASes. While we observe large day-to-day variations during the month (logarithmic y-axis), it is encouraging that the average node generate on average less than 5K queries per day and the number of subprefix hijack alerts (Figure 12(a)) and prefix hijack alerts (Figure 12(b)) scales very nicely with the number of members. For example, the average number of alerts changes from 2,527 to 4,133, and from 355 to 462 for the two types, respectively, as the number of collaborating ASes increases from 10 to 271.

Keeping in mind that the storage overhead and number of prefixes (Section 3.3) that each holder node is responsible for decreases in inverse proportion to the number of alliance members, we note that the queries processed per alliance node remains roughly constant, as this directly cancels the linear increase in the number of original queries generated by the entire alliance. In fact, with sub-linear increase in the number of alerts, it can be argued that the overhead per node decreases with growing alliance sizes.

4.3. Scale-based and size-based analysis

Several studies have suggested that there are significant benefits to deploying hijack detection mechanisms on several large ASes across the world [29, 11, 30]. Similarly, our results in the previous section also show how collaboration among a few ASes spread across the globe would have helped raise many useful alerts during the China Telecom incident. However, global deployment that spans multiple geographic regions and jurisdictions is non-trivial and may not always be practical due to political and economic reasons. Even the choice of technology can becomes an issue limiting the global collaboration. For example, although PrefiSec is designed to allow the use of any encryption algorithm (e.g., leveraging TLS/SSL for per-hop connections or a common shared key) for the sharing of information between participants, the particular choice and information shared may impact the potential membership as network operators may be under different laws and regulation regarding what encryption algorithms must



(a) Number of participating ASes (b) Size of participating ASes

Figure 13: Average number of alerts raised when global ASes collaborate the day of the China Telecom incident.

be used here. In some cases, it may be easier to at least initially push or incentivize the deployment of systems of this type within a geographic region such as the US or EU. For example, governmental legislation or other regional mechanisms could be used to push or incentivize agreements between ASes within the region.

To understand the impact of such regional restrictions, we next compare and evaluate the benefits and drawbacks of deploying PrefiSec regionally versus globally. We first present the results for the scale and size aspects from a global perspective. This is followed by the results for regional perspective where PrefiSec is assumed to be used by ASes in specific regions such as North America (NA), European Union (EU), and "Rest of the world" (all the ASes that do not belong to NA or EU region).

As a baseline, we first present results for when the collaborating ASes are selected globally. Figure 13(a) shows the number of alerts raised for both "new prefixes" (possible subprefix hijacks) and "new prefix origins" (possible prefix hijacks) announced during the incident (on April 8) as a function of number of collaborating ASes. We also include separate lines for the number of alerts of these two types raised due to announcements made by China Telecom.

We see that the number of alerts for possible prefix hijacks increases with the number of collaborating ASes, and that 40,575 alerts (for both prefix and subprefix hijacks) are raised during the day of the attack if all the nodes collaborate. With the exception of a few "new prefixes" and "new prefix origins", almost all alerts are due to the China Telecom announcements associated with the incident.

Only a few ASes are needed to detect the majority of the subprefix hijacks ("new prefixes"). This result can be explained by subprefixes being propagated to almost all ASes due to more specific prefixes being preferred. For prefix attacks ("new origin") additional ASes are much more beneficial, with some diminishing returns after reaching 40 ASes. This happens because ASes during these instances become divided into two groups: ASes that continue routing to the victim network and ASes that choose to route to the attacker network.

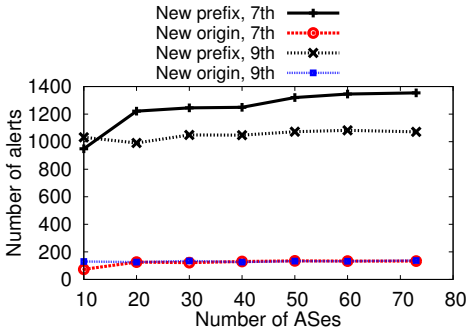


Figure 14: Average number of alerts raised when global ASes collaborate the day before (April 7) and after (April 9) the incident.

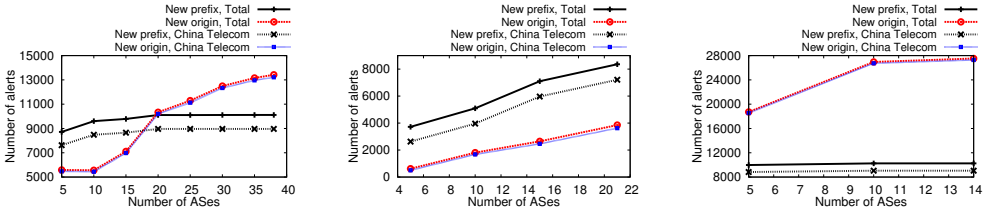
Thus, additional collaborating ASes increase the chance that conflicting origins will be detected and hijack alerts will be raised.

Figure 14 puts the above numbers in perspective, showing the number of alerts for the days before and after the attack. In addition to being orders of magnitude lower than during the day of the incident, the flatter "new origin" curves suggest that the "new origin" announcements during these days propagated somewhat further than the China Telecom announcements.

Figure 13(b) shows the number of alerts as a function of the degree threshold to be included in the alliance. For every threshold, ten ASes with a degree of at least X are selected at random. Here, the largest (right-most) displayed threshold is picked so that the selection set includes exactly ten ASes, and the following decreasingly smaller thresholds are picked so as to roughly double the selection set for each threshold going from right to left in the figure. The degree threshold of one is included as a reference point.

The figure shows that the number of alerts for the China Telecom incident is higher when the degree threshold is small, and the number of alerts is quite low when large ASes collaborate. This is a very interesting observation as much prior work has suggested collaboration between the largest ASes gives better security gains, but it can be partially[2] explained by most of the high degree ASes being NA-based. For example, of the ASes with a degree greater than 1,174, all but one (i.e., nine out of ten) are NA-based, and when the threshold is 646, there are 18 NA-based and two EU-based. However, these NA-based ASes do not have as good a vantage point of the China-based incident, with only a subset of the paths propagating to these ASes. With a lower degree threshold more ASes from outside NA and EU will be included, improving the results. This illustrates that the vantage points offered by global collaboration can be more valuable to the prefix hijack detection than having only the large ASes collaborate. Similarly, multi-hop BGP peering can also help. The detection

[2]Additional explanation will be provided in the next subsection.



(a) North America (NA) (b) European Union (EU) (c) Rest of the world

Figure 15: Number of alerts during the day of the incident (April 8, 2010) for different sizes of regional collaborations.

numbers for subprefix attacks ("new prefixes") are less dependent of the AS degree (size) and locality; again, indicating their wider propagation.

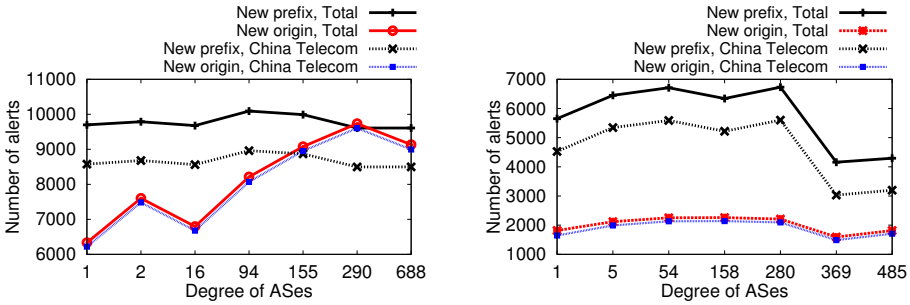
4.4. Location-based analysis

We now discuss the benefits of regional collaboration for hijack detection. Figure 15 shows the number of alerts as a function of number of ASes for different regions. For all of the three regions (NA, EU, and "rest of the world"), the number of alerts increases as more ASes share information. If all NA-based ASes collaborate there are 22,178 alerts (13,214 "new origin" and 8,964 "new prefix"). Sharing among all EU-based ASes raises 10,829 alerts (3,620+7,209) and sharing among all the ASes in the "rest of the world" category would raise 36,328 alerts (27,280+9,048). Whereas the sub-prefix detection ("new prefix") is similar for the different regions, the differences in total alerts are substantial. For example, despite there being far fewer ASes in the "rest of the world" category, this category has the highest detection rate. The main reason for this is that many of these ASes have more vantage points closer to China Telecom than NA-based and EU-based ASes may have, and therefore have better visibility of the route announcements made by China Telecom. This observation mirrors the insights provided by our evaluation of BGP hijack prevention mechanisms [31] that show that ASes deploying protection mechanisms close to the attacker provide the best protection.

While none of the regional collaborations performs as well as global collaboration, the value of regionally deployed solutions should not be underestimated, especially as there is no solution that has seen widespread deployment yet. These results show that careful regional deployment, possibly with a few complementing ASes from other regions, may provide a significant step in the right direction.

Figures 16(a) and 16(b) show the number of alerts as a function of the degree threshold for regional collaborations in NA and EU, respectively. As for the global results, for each degree threshold, we randomly pick ten ASes per alliance.

We again observe a stronger degree (size) dependence for prefix hijack detection ("new origins") than for subprefix hijack detection ("new prefixes"). While the large ASes in NA in general provide more alerts than the smallest ASes in



(a) North America (NA) (b) European Union (EU)

Figure 16: Impact of the size of the participating ASes on the number of alerts. For each threshold size we choose 10 ASes with degree equal or greater than the applied threshold.

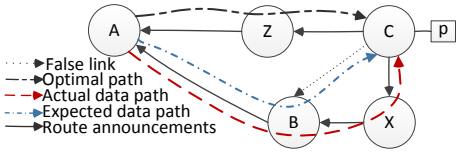


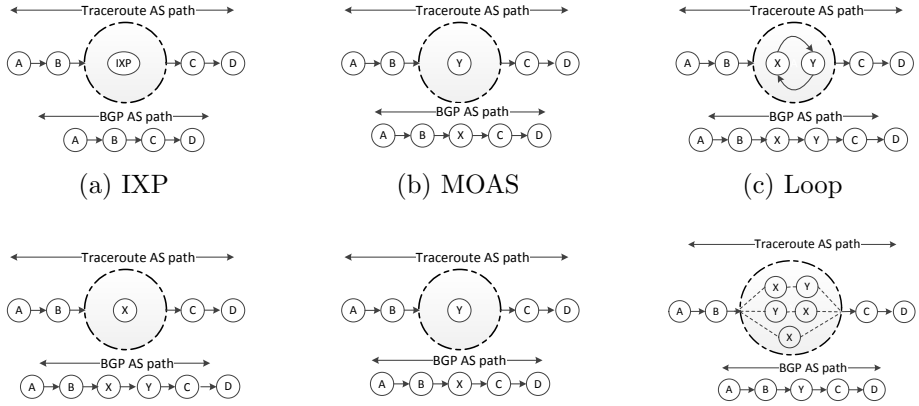
Figure 17: Detecting route inconsistencies.

NA, it is very interesting that the very top ASes see a drop in the number of alerts they raise. It is also interesting that the large ASes in EU detect fewer attacks than the smaller ASes in EU. As the above ASes are in the same region, our previous explanations (in Section 4.3) regarding the relative differences in coverage seen by ASes in *different regions* no longer apply here. In the *same region*, the size-based differences may instead be related to the standard route export policy. In particular, malicious routes (learnt from a peer or provider) are typically exported only to customers. Therefore, malicious routes learnt by mid-tier ASes may not reach their providers (typically large ASes).

5. Interception attack

One of the harder problems with BGP security is the detection of interception attacks [1, 2]. Figure 17 shows an example. Here, AS B announces that it is one hop away from C, although in reality, it is not connected to C. This announcement will not result in any prefix origin triggers, but may still allow B to intercept traffic on its way to C.

As of today there is no straightforward way to automatically detect interception attacks. Instead, network owners must typically manually analyze and resolve suspicious inconsistencies between announced BGP AS-PATHs and the actual data paths. This section describes how PrefiSec can be used to reduce the number of suspicious inconsistencies.



(a) IXP (b) MOAS (c) Loop

(d) AS hop missing (e) Alias (f) Sibling

Figure 18: Legit reasons for path discrepancies.

5.1. Policy overview

We envision that members will maintain a history of the announced AS-PATHs, and evaluate any newly observed path-prefix pairs for inconsistencies. At such times, the member node (1) performs a traceroute to an IP address within the prefix, (2) uses the prefix registry to create a traceroute AS path [32], and (3) compares the announced AS-PATH (control-plane information) with the traceroute AS path (data-plane information). If the traceroute AS path does not match the announced AS-PATH, (4) the node uses information maintained by the AS registry regarding legit path discrepancy reasons. Finally, if no legit reason is found, (5) the node raises an alert and informs the appropriate AS and prefix holder nodes.

5.2. Legit path discrepancy reasons

To reduce the number of false alerts, it is important to keep track of legit reasons for suspicious path discrepancies between the announced AS-PATHs and the actual data paths. Figure 18 summarizes some common legit reasons for such differences [32]. We next describe how the AS registry can maintain information about such reasons.

IXP cases (Figure 18(a)): Internet eXchange Points (IXPs) [32] may cause extra hops in the traceroute path, not seen in the announced AS-PATHs. Extending the approach by Mao et al. [32], nodes that detect an extra AS hop X can report the ASes before and after X to the holder of X. This node can then calculate the number of unique ASes appearing just before and after X, referred to as the fan-in and fan-out factor, respectively. If these factors are greater than some threshold, the holder can classify X as an IXP.

MOAS cases (Figure 18(b)): In certain cases we may observe that AS X in the AS-PATH is replaced by AS Y in the traceroute path. Such replacement is

common when the prefix is originated by multiple ASes (MOAS). We note that such MOAS cases are an artifact of mapping from an IP address to AS and not a result of a routing anomaly and can be identified by holder nodes. Holders in the prefix registry can be informed about multiple co-origins, as described for IXPs, which could inform the AS holders about these relationships. Alternatively, the AS holders themselves can keep track of replacements reported by member nodes.

Loop cases (Figure 18(c)): Some traceroute paths exit and enter an AS more than once [32]. For example, an announced AS-PATH {A,B,C,D} may have a corresponding traceroute path {A,B,C,B,C,B,C,D}. These cases do not require any additional information from the AS registry.

Missing-hop cases (Figure 18(d)): Occasionally, an AS hop seen in the AS-PATH is not observed in the traceroute path. For example, in Figure 18(d), AS Y is missing in the traceroute path. This can occur for reasons such as routers in Y not responding to traceroute queries or using IP addresses from their neighbors. This case typically does not require any additional AS registry information, although it would be easy to add more AS information to the holder.

Alias cases (Figure 18(e)): When an AS X in the AS-PATH is replaced by an AS Y in the traceroute path, it may be due to a router having IP addresses from two different ASes on its interfaces. Such an IP address, called an alias address, may arise due to third-party address issues [33]. The alliance nodes can use existing third-party address detection methods [33], and report its findings to the holder node of the replacement AS hop Y in the AS registry, which can apply a threshold-based policy on the number of occurrences required for X and Y to be classified as an alias pairing.

Sibling cases (Figure 18(f)): Other potential causes for valid discrepancies are route aggregation and sibling ASes, owned by the same organization [32]. Figure 18(f) illustrates a case in which the AS-PATH is $\{A,B,Y,C,D\}$ and ASes X and Y are sibling ASes. In the traceroute path we may observe Y being replaced by any of the following: $\{X,Y\}$, $\{Y,X\}$, or $\{X\}$. When an alliance node encounters such a case, it will report the AS-hop before and after the two-hop segment $\{X, Y\}$ in the traceroute path to the holder nodes of both X and Y. Similar to in the IXP case, if the fan-in and fan-out exceeds a threshold, the holder node detects a sibling relationship [32], and can inform the other holder.

5.3. Case-based analysis

We next consider how an AS can use the information provided by PrefiSec to identify suspicious and non-suspicious path inconsistencies. For this evaluation, we use measurements from three public RouteViews monitors and three nearby public traceroute servers, each pair hosted by Global Crossing (AS 3549), Telstra (AS 1221), and Hurricane Electric (AS 6939). These servers are located in Palo Alto (CA), Sydney (Australia), and San Jose/Livermore (CA).

Traceroutes: As with the prefix hijack detection overhead, great reductions in the number of traceroutes that must be executed can be achieved using a

Table 1: Reduction in the number of traceroutes.

Extra history (hours)	Server		
	Telstra	Global	Hurricane
2h	25.8%	25.1%	24.3%
4h	31.4%	32.6%	33.5%
8h	34.9%	38.2%	38.9%
16h	47.4%	45.6%	47.1%
24h	49.9%	48.6%	50.4%
48h	53.6%	51.9%	53.4%

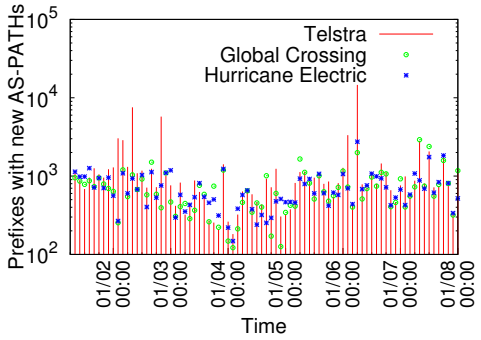


Figure 19: New routes observed (Nov. 1-7, 2012).

short history of previously seen AS-PATHs. Table 1 shows the reduction in the number of traceroutes to execute, when considering multiple RIB snapshots (each two hours apart), at each of the three servers. We observe diminishing returns, with most of the advantages obtained using a relatively small history. In the following we use a 24-hour history (49% reduction).

Figure 19 shows the number of traceroutes that would have to be executed, as a function of time, during the first week of November 2012. The three servers observe similar variations in the number of traceroutes they need to perform at each instance during this period, although the pairwise Pearson correlation factors (TG: 0.403; TH: 0.604; HG: 0.661) suggest that the correlation is only moderate.

Path comparison: For our analysis, we first convert the IP-level traceroutes to AS-level traceroutes. While our prefix registry is designed to provide this mapping, for the purpose of our evaluation, we used the Cymru whois database. Nodes not responding to ICMP queries are considered as wild-cards ($*$) between the neighboring ASes.

As mentioned, there are many valid reasons why a traceroute path will not be a *direct match* to the announced AS-PATH. First, the IP-to-AS mapping could be incorrect or out-of-date. While our prefix registry will help, there is no 100% up-to-date information source for this mapping. Second, it may not be possible to map all routers along the paths, as some routers (wild-card nodes) may have disabled ICMP, or traceroute servers (in this case out of our

Table 2: Path comparison analysis (Nov. 1-7, 2012).

	Telstra	Global	Hurricane
Announcements	$3.6 \cdot 10^7$	$3.5 \cdot 10^7$	$3.6 \cdot 10^7$
Traceroutes (new route)	102,689	63,434	60,628
No data (new route)	506	60,045	3,200
Successful traceroutes	102,183	3,389	60,627
Direct matches	12,387	704	16,374
Subset matches	62,952	947	13,267
IXP matches	2,672	11	30,071
MOAS matches	309	NA	445
Loop matches	2,271	108	1,021
Missing hop matches	2,730	689	6,886
Alias matches	11,650	276	7,020
Sibling matches	1,764	27	243
Past matches	4,209	363	1,916
Future matches	487	23	515
Unresolved traceroutes	3,333	244	9,464
Unresolved triples	539	82	1,422

administration) may terminate at some timeout value. We refer to queries that matched all observed ASes as *subset matches*. This scenario was common for the Telstra servers.

Third, our AS registry keeps track of AS relationship information about the six legit reasons for route anomalies described in Section 5.2. We call paths that match after applying each such condition (sequentially): *IXP matches*, *MOAS matches*, *loop matches*, *missing hop matches*, *alias matches*, and *sibling matches*. To approximate the conditions that a large-scale system would see, we populate the AS registry using public data, including known IXP prefixes [34], aliases from the iPlane project [35], and sibling relationship data from CAIDA [36]. MOAS cases are identified using IP-to-AS mappings from the whois Cymru database.

Finally, the announced AS-PATHs and traceroute paths may change over time, and may not always be in sync. Using a 2-hour window of path announcements seen at the server, we identify *past matches* and *future matches*. Past matches captures fluctuations in AS-PATHs. The future matches results from cases in which the announced path changes have not yet propagated all the way to the monitor server. If employed, the future match policy would of course require a time delay (e.g., two hours) before classifying a path.

Table 2 provides a breakdown of the traceroutes that we performed during the first week of Nov. 2012. We applied each rule in order, such that only traceroutes that did not match the previous criteria were considered for each new row. The *no data* cases correspond to cases in which the public traceroute server did not respond to our traceroute queries. The high number of such queries to the Global Crossing server is due to server limitations, but is not expected to have introduced biases affecting our results.

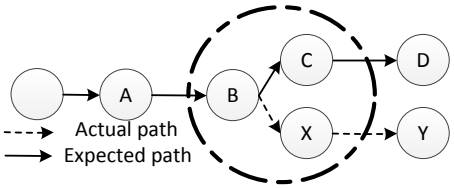


Figure 20: ASes involved in the anomaly

Table 3: Redundancy in unique triples.

				Redundancy in triples		
	Server	Traceroute	Triples	History	Sharing	Hybrid
Week 1	Telstra	66,985	372	–	12%	–
	Global	55,144	292	–	15%	–
	Hurricane	–	–	–	–	–
Week 2	Telstra	102,689	539	32%	14%	41%
	Global	63,434	82	27%	26%	38%
	Hurricane	60,628	1,422	4%	6%	7%
Week 3	Telstra	77,012	492	45%	16%	54%
	Global	–	–	–	–	–
	Hurricane	56,518	1,244	35%	6%	39%

We note that the number of traceroutes that we could not automatically confirm (3,333; 244; and 9,464) is much smaller than the original number of successful traceroute queries (102,183; 3,389; and 60,627). We also see that IXP, alias, MOAS, and sibling matches are each responsible for a significant portion of the reduction of unique candidate announcements, validating the importance of the AS registry.

Further reductions: To further reduce the candidate cases to consider, we identify unique triples [2]. Referring to Figure 20, such a triple consists of (i) AS B that is responsible for the anomaly, (ii) AS C that B claims it will use, and (iii) AS X over which it actually forwards the packet. The last line in Table 2 shows that grouping into triples allows a 65-85% reduction in the number of cases to consider, as the reasons for these anomalies are often the same.

For the remaining candidate anomalies, nodes can contact the holders of AS B in the triple. If the holder node has seen this anomaly before and has found no problem, then it can respond that the anomaly is indeed benign. Otherwise, the holder node may need to do additional analysis.

The holder nodes are in a great position to take advantage of aggregate information. Table 3 shows the summary statistics for three basic aggregation and history-based approaches. The history column shows the percentage of triples that have also been observed by the same server in past weeks. The sharing column shows the percentage of triples that are also seen by at least one of the other servers (one or two, depending on the week). Finally, the

hybrid column shows the percentage of triples that apply to at least one of the rules. We see that with just three members the number of triples that must be processed can be reduced by up to 43% (week 3). Clearly, there are significant advantages to maintaining history at the holder nodes, and sharing information across organizations.

We note that none of our policies or mechanisms thus far has relied on any knowledge of AS-to-AS relationships. An AS also wanting to investigate the *cause* for the remaining unresolved anomalies, may want to make use of AS-to-AS relationships such as customer-to-provider (C2P), peer-to-peer (P2P), and provider-to-customer (P2C) [13]. Unfortunately, most ASes do not want to share their peering policies. To incorporate a classification of AS-to-AS relationships into the AS registry, future work could therefore include the design and evaluation of a distributed version of the valley-free [32] classification method by Shavit et al. [37].

Overhead discussion: Referring back to our interception detection and alert notification policy (Section 5.1), we note that all traceroutes (step 1) and path comparisons (steps 3) are done locally at the detecting node. Holder nodes are responsible for raising alerts and working with the identified triples (step 5). The overlay is only invoked when doing IP-to-AS mappings (step 2) and using AS relationships to reduce the number of mismatches (step 4). Of these, step 2 involves the most communication. For example, during the first week of Nov. 2012, our policy performed 14,670 traceroutes per day at the Telstra server and only 3,907 of these needed to be considered for further reductions after removing direct and subset matches. In contrast, 265,805 IP addresses where observed per day in the traceroutes.

Fortunately, out of the observed IP addresses, we observed only 8,675 unique IP addresses per day on average. Furthermore, if considering unique /24 prefixes, we could further reduce the number of queries to 3,967, and finally if also allowing a 24 hour rolling window of observed /24 prefixes this number is reduced to 2,787. The overlay should easily handle these lookups, and the processing load on each holder node does not change with the number of members (as the increased total load is distributed over more holders). The only increase in load that results from increasing membership is due to longer Chord routes. However, as discussed in Section 3.2, these overheads can easily be minimized based on if storage or forwarding costs are the primary bottleneck.

5.4. Location-based discussion

We now discuss the implications of regional collaboration with regards to our interception detection policy in PrefiSec.

Alert rates: Since we have very limited ASes for this analysis, conclusive results are difficult to derive. However, assuming collaboration between the two NA-based ASes (GlobalCrossing and Hurricane Electric), the number of triples that must be processed reduces by up to 23% (63 triples still need to be resolved) for GlobalCrossing and 1.3% (1403 triples still need to be resolved) for Hurricane Electric. When collaboration with Telstra is also considered these numbers are 27% and 4% for Global Crossing and Hurricane Electric, respectively. These

numbers suggest that there are cases were local ASes can both complement each other and provide overlapping information.

Populating information in AS registry: The interception detection policy in PrefiSec depends on the information about ASes that the AS registry learns from the participating ASes. When few nodes are collaborating, as in the case of regional collaboration, some cases may be more difficult to detect. For example, to detect IXPs, a minimum threshold of the fan-in and fan-out factor is required before an AS can be classified as an IXP. If the number of affected participating ASes is below this threshold, the IXP will not be detected. Such cases will impact the outcome of the path comparison analysis that we have presented.

The information needed for path comparison often does not change frequently. For example, a set of IXPs and their prefixes should be fairly stable and it is not expected that new IXPs will appear frequently. Therefore, ASes collaborating regionally can potentially complement their own information and the information in their registry with externally generated IXP information. For example, IXP-specific data can be extracted from sources such as Packet Clearing House [38] and PeeringDB [39] and used for path comparison analysis. Similar to IXP related information, other information such as MOASes, aliases, and sibling ASes is also quite stable over a period of time and information retrieved externally can be used for path comparison analysis. Thus, the inherent limitation from small number of ASes in regional collaboration can be mitigated. Cross-validation over many complementing information sources can also be used to strengthen the belief in observed instances. In general, however, we believe it is important for organizations to actively and proactively collect information for such cross-validation. PrefiSec provides an important tool for organization in this regard.

Overhead: Having regional collaboration instead of global collaboration typically results in shorter paths between participating ASes. Therefore, referring to Section 3.3, we would typically see fewer IP-level hops per chord lookup. Thus, one can expect lower overhead and quicker resolution of various queries and reports generated by the nodes in the overlay network when regional deployment is considered.

6. Prefix Monitoring

In addition to routing-aware information, our prefix registry is designed to help organizations share anomaly alerts and effectively aggregate (often sparse) information about a wide range of attacks associated with edge networks and their prefixes. Shining the light on the organizations and network owners themselves, has many advantages. For example, using timely reports from other alliance members, the network owners can police miscreant activity within their own networks, clean up their network footprint, and ensure that compromised machines do not cause prolonged harm [8]. The use of prefixes also allows the system to scale effectively, and helps capture effects of subnetwork-aware IP-

spoofing [4] and address migration due to dynamic address allocation. We use four basic examples to illustrate the power and generality of the framework.

Scanning attacks: To detect scanning, organizations typically monitor the incoming (and outgoing) traffic using intrusion detection systems (IDS) and classify each connection as either good or bad. Allman et al. [3] propose a set of heuristics to classify hosts based on the mix of good/bad connections. Our system is well suited to implement generalizations of such classification policies [4], in which prefixes are evaluated rather than hosts.

Spam server activity: Similarly, organizations can monitor SMTP traffic from non-legit servers and report such activity to the holder nodes. This information can then be used to inform the origin owner of the prefix of suspicious activity, and/or provide organizations with information about the general trust level associated with different prefixes. We note that this approach naturally extends to many other activities, including the presence of botnet servers.

Cross-class detection: It is common for malicious hosts to alternate between different malicious behaviors [6]. The holders have access to information related to many different types of attacks associated with a prefix, and is therefore in a good position to perform cross-class detection.

Attack correlation: Finally, networks are typically not attacked at random and often attacks are not isolated [5]. Targeted networks might therefore often benefit from additional information sharing and collaboration. Holders can act as matchmakers, informing victim networks about correlated networks that see similar attacks.

7. Related work

DHTs and DHT-based applications: Distributed Hash Tables (DHTs) such as Chord [10], have been used to improve the scalability for a wide range of distributed systems, including for co-operative web caching [40] and publish-subscribe systems [41]. To the best of our knowledge, this is the first work to allow the use of IP-prefix ranges.

Collaboration: Collaboration has been shown to help protect against different types of edge-network attacks [42], including DDoS attacks, system intrusions, and scanning. With these approaches, Intrusion detection system (IDS) monitors typically share data plane information and malicious IP addresses [42]. Collaborative fault detection has also been proposed for BGP [43]. With NetReview [43] BGP routing messages are recorded in tamper-evident logs that can be shared with others. Our systems are complementary, as NetReview could be used to share richer information to carefully analyze suspicious activity identified using our system, for example.

Central solutions: Both BGPMon[3] and Team Cymru[4] collect routing information from distributed monitors, and create alerts/summary reports about

[3]BGPMon, `http://www.bgpmon.net/`, July 2016.
[4]Team Cymru, `http://www.team-cymru.org`, July 2016.

routing anomalies to which organizations can subscribe. Although Team Cymru provides a DNS-based lookup service for origin ASes, prefixes, and allocation dates, the service and all processing is centralized under a single administrative domain. In this paper we propose and evaluate a distributed solution.

Crypto-based architectures: The Resource Public Key Infrastructure (RPKI) [44] builds a formally verifiable database of IP addresses and AS numbers, and can be used to verify that the AS originating a prefix is authorized. While some recent works (e.g., [45]) have identified significant issues with the hierarchical RPKI management and the control it can give some entities (e.g., RIRs) on the global Internet routing (which can have significant political and business implications), RPKI also has many nice features, including incremental deployment (after some router software updates) and the ability to block hijacks. Proposals such as BGPsec [46] extend RPKI, to protect the AS path attribute of BGP update messages. Rather than blocking specific attacks, PrefiSec provides a scalable, fully distributed architecture for sharing of information that can be used to protect against many types of attacks.

Reverse DNS: ROVER [47] is a complementary approach to cryptographically secure BGP route origin, which builds on DNSSEC [48] and reverse DNS services. Despite years in development, less than 0.7% of .com and .net domains had signed up for DNSSEC by July 2016.[5] Requiring manual configuration, the future adoption rate of ROVER is unclear. In addition, recent revelations about government sponsored online surveillance have raised concerns regarding systems that require centralized root key distribution.

Hijack detection: There has been no large-scale deployment of crypto-based solutions [1]. Instead, both data-plane based [49, 50] and control-plane based [11, 12] techniques have been proposed for anomaly detection in BGP routing. As explained in Section 4, we extend existing control-plane based protocols (PG-BGP [11], PHAS [12]) for prefix ownership. Similar to Ballani et al. [2], we combine data-plane and control-plane information to detect and classify routing anomalies, but, in contrast, consider the more general case in which the anomaly can occur in the middle of the path. The combination of control-plane and data-plane data have been used to evaluate the accuracy of AS-level topology inference [51] and detecting hidden areas of the Internet [52]. These works have provided additional insights to our design.

Locality and size aspect: While partial deployment of BGP security mechanisms has been considered in prior literature [53, 54, 30, 11], geographic location of participants has almost always been ignored. Instead, carefully selected ASes have typically been used to demonstrate the potential of the individual techniques. For example, Avramopoulos et al. [30] demonstrate good protection of participants' outgoing and incoming traffic using only the top-5 tier-1 ASes in the world. Others have relied on the top-tier ASes to demonstrate the effectiveness of PG-BGP [11], path validation protocols such as S-BGP and BGPSec [54], and incentive strategies for deployment of S*BGP [53]. None of

[5] Verisign labs scoreboard, `http://scoreboard.verisignlabs.com/`, July 2016.

these works considers the impact of locality of the ASes that are deploying the security mechanisms.

We are not the first to study the impact of the AS-neighbor degree of participating ASes [55] or the effect of the number of participating ASes [29]. For example, Suchara et al. [55] analyze security gains as a function of increasing the AS-neighbor degree of the ASes that use a BGP security mechanism that filters malicious routes. Similar to our results, they find significant benefits to deploying the mechanism at high-degree ASes at the core of the Internet. Gersch et al. [29] analyze the effect of increasing number of ASes using attack prevention techniques. Their results nicely show how the average number of polluted ASes decreases with increasing number of participating ASes (with higher degrees). Again, none of these works consider the geographic region that each AS maps to. This is a factor that can be important when it comes to legislation and other political incentives to deploy the proposed routing security mechanisms.

8. Conclusions

This work presents the design and data-driven overhead analysis of PrefiSec, a distributed system that provides scalable and effective sharing of network information, for the purpose of helping organizations detect and protect against prefix/subprefix attacks, interception attacks, and a wide range of edge-based attacks. We present a novel distributed solution, which maintains information about the activity associated with blocks of IP addresses (prefixes) and autonomous systems (AS). Our solution extends Chord [10], leverages unique properties of CryptoPAN [20], and implements new scalable mechanisms and policies for efficient information sharing. Using public wide-area BGP-announcements, traceroutes, and simulations, we show that the system is scalable with limited overhead. The system helps participants improve their own security and keep the network footprints of their own networks relatively clean from miscreant activities. Our distributed mechanisms infer AS relationships from publically available information, including public route announcements, and participating ASes are not expected to share any private information about their own networks and AS relationships. Of course, further improvements can be achieved if ASes also share some private information.

Our results show that collaboration between a small number of ASes can be sufficient to detect new prefixes (possible sub-prefix hijacks). The results also show that having additional collaborating ASes is beneficial for detection of new origin announcements (possible prefix hijacks), with diminishing returns after a certain threshold. An interesting observation is that AS relationships can cause collaboration between large ASes in a particular region to provide lower hijack detection rates if the attacks originate from other regions compared to hijack detection provided by collaboration between mid-sized ASes in the same region. Future work includes the design and evaluation of sharing policies and incentive mechanisms, leveraging our incentive-based hierarchy extension.

Acknowledgement

This work was supported by the Swedish national graduate school in computer science (CUGS).

References

[1] K. Butler, T. Farley, P. McDaniel, J. Rexford, A survey of BGP security issues and solutions, Proceedings of the IEEE 98 (2010) 100–121.

[2] H. Ballani, P. Francis, X. Zhang, A study of prefix hijacking and interception in the Internet, ACM CCR 37 (2007) 265–276.

[3] M. Allman, V. Paxson, J. Terrell, A brief history of scanning, in: Proc. ACM IMC, 2007.

[4] M. Arlitt, N. Carlsson, P. Gill, A. Mahanti, C. Williamson, Characterizing intelligence gathering and control on an edge network, ACM TOIT 11 (2011) 2:1–2:26.

[5] A. Ramachandran, N. Feamster, Understanding the network-level behavior of spammers, in: Proc. ACM SIGCOMM, 2006.

[6] H. Ringberg, A. Soule, M. Caesar, Evaluating the potential of collaborative anomaly detection, Tech. rep. (2008).

[7] Symantec, 2013 Internet Security Threat Report (2013).

[8] N. Carlsson, M. Arlitt, Leveraging organizational etiquette to improve Internet security, in: Proc. IEEE ICCCN, 2010.

[9] S. Katti, B. Krishnamurthy, D. Katabi, Collaborating against common enemies, in: Proc. IMC, 2005.

[10] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, H. Balakrishnan, Chord: A scalable peer-to-peer lookup service for Internet applications, in: Proc. ACM SIGCOMM, 2001.

[11] J. Karlin, S. Forrest, J. Rexford, Pretty Good BGP: Improving BGP by Cautiously Adopting Routes, in: Proc. IEEE ICNP, 2006.

[12] M. Lad, , D. Pei, Y. Wu, B. Zhang, L. Zhang, PHAS: A prefix hijack alert system, in: Proc. USENIX Security Symp., 2006.

[13] R. Hiran, N. Carlsson, P. Gill, Characterizing large-scale routing anomalies: A case study of the china telecom incident, in: Proc. PAM, 2013.

[14] R. Hiran, N. Carlsson, N. Shahmehri, PrefiSec: A distributed alliance framework for collaborative BGP monitoring and prefix-based security, in: Proc. ACM CCS Workshop on Information Sharing and Collaborative Security, Scottsdale, AZ, 2014.

[15] S. Goldberg, Why is it taking so long to secure Internet routing?, ACM Queue 12 (8) (2014) 327–338.

[16] A. Peterson, Researchers say U.S. internet traffic was re-routed through Belarus.That's a problem. (Nov. 2013).
URL http://www.washingtonpost.com/blogs/the-switch/wp/2013/11/20/researchers-say-u-s-internet-traffic-was-re-routed-through-belarus-thats-a-problem/

[17] Dyn Research, The new threat: Targeted internet traffic misdirection (Nov. 2013).
URL http://research.dyn.com/2013/11/mitm-internet-hijacking/

[18] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, J. van der Merwe, The case for separating routing from routers, in: Proc. ACM SIGCOMM FDNA, 2004.

[19] R. Fernando, S. Stuart, BGP Monitoring Protocol, IETF (2012).

[20] J. Fan, J. Xu, M. H. Ammar, S. B. Moon, Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme, Computer Networks 46 (2004) 253–272.

[21] A. Gupta, B. Liskov, R. Rodrigues, Efficient routing for peer-to-peer overlays, in: Proc. NSDI, San Francisco, CA, 2004.

[22] P. Garca, C. Pairot, R. Mondjar, J. Pujol, H. Tejedor, R. Rallo, PlanetSim: A new overlay network simulation framework, in: Proc. SEM, 2004.

[23] J. Czyz, M. Allman, J. Zhang, S. Iekel-Johnson, E. Osterweil, M. Bailey, Measuring IPv6 adoption, in: Proc. ACM SIGCOMM, 2014.

[24] G. Huston, IPv6 cidr report, Statistics report (Nov. 2016).
URL http://www.cidr-report.org/v6/as2.0/

[25] A. Lutu, M. Bagnulo, C. Pelsser, O. Maennel, Understanding the reachability of IPv6 limited visibility prefixes, in: Proc. PAM, 2014.

[26] IAB and IESG, IAB/IESG Recommendations on IPv6 Address Allocations to Sites, RFC 3177, Informational, IETF (2001).
URL https://tools.ietf.org/html/rfc3177

[27] T. Narten and G. Huston and L Roberts, IPv6 Address Assignment to End Sites, RFC 6177, Best Current Practice, IETF (2011).
URL https://tools.ietf.org/html/rfc6177

[28] C. Duma, N. Shahmehri, G. Caronni, Dynamic trust metrics for peer-to-peer systems, in: Proc. PDMST, 2005.

[29] J. Gersch, D. Massey, C. Papadopoulos, Incremental deployment strategies for effective detection and prevention of BGP origin hijacks, in: Proc. IEEE ICDCS, 2014.

[30] I. Avramopoulos, M. Suchara, J. Rexford, How small groups can secure interdomain routing, Tech. rep., Princeton University (2007).

[31] R. Hiran, N. Carlsson, N. Shahmehri, Does scale, size, and locality matter? evaluation of collaborative BGP security mechanisms, in: Proc. IFIP Networking, 2016.

[32] Z. M. Mao, J. Rexford, J. Wang, R. H. Katz, Towards an accurate AS-level traceroute tool, in: Proc. ACM SIGCOMM, 2003.

[33] P. Marchetta, W. de Donato, A. Pescapé, Detecting third-party addresses in traceroute traces with IP timestamp option, in: Proc. PAM, 2013.

[34] B. Augustin, B. Krishnamurthy, W. Willinger, IXPs: mapped?, in: Proc. ACM IMC, 2009.

[35] H. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, A. Venkataramani, iPlane: An information plane for distributed services, in: Proc. OSDI, 2006.

[36] M. Luckie, B. Huffaker, A. Dhamdhere, V. Giotsas, kc claffy, AS relationships, customer cones, and validation, in: Proc. IMC, 2013.

[37] Y. Shavitt, E. Shir, U. Weinsberg, Near-deterministic inference of AS relationships, in: Proc. ConTEL, 2009.

[38] Packet Clearing House, Internet exchange directory (2016).
URL `https://prefix.pch.net/applications/ixpdir/`

[39] PeeringDB, PeeringDB facilitates the exchange of information related to peering (2016).
URL `https://www.peeringdb.com/`

[40] S. Iyer, A. Rowstron, P. Druschel, Squirrel: a decentralized peer-to-peer Web cache, in: Proc. ACM PODC, 2002.

[41] R. Baldoni, C. Marchetti, A. Virgillito, R. Vitenberg, Content-based publish-subscribe over structured overlay networks, in: Proc. IEEE ICDCS, 2005.

[42] C. V. Zhou, C. Leckie, S. Karunasekera, A survey of coordinated attacks and collaborative intrusion detection, Computers and Security 29 (2010) 124–140.

[43] A. Haeberlen, I. Avramopoulos, J. Rexford, P. Druschel, NetReview: Detecting when interdomain routing goes wrong, in: Proc. NSDI, 2009.

[44] M. Lepinski, S. Kent, An Infrastructure to Support Secure Internet Routing, RFC 6480 (Informational) (2012).

[45] D. Cooper, E. Heilman, K. Brogle, L. Reyzin, S. Goldberg, On the risk of misbehaving RPKI authorities, in: Proc. ACM HotNets, 2013.

[46] M. Lepinski, BGPSEC protocol specification, IETF (2013).

[47] J. Gersch, D. Massey, ROVER: Route origin verification using DNS, in: Proc. IEEE ICCCN, 2013.

[48] R. Arends, R. Austein, M. Larson, D. Massey, S. Rose, Protocol Modifications for the DNS Security Extensions, RFC 4035 (Proposed Standard) (2005).

[49] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, R. Bush, Ispy: detecting IP prefix hijacking on my own, ACM CCR 38 (2008) 327–338.

[50] C. Zheng, L. Ji, D. Pei, J. Wang, P. Francis, A light-weight distributed scheme for detecting IP prefix hijacks in real-time, in: Proc. ACM SIGCOMM, 2007.

[51] Y. Zhang, R. Oliveira, Y. Wang, S. Su, B. Zhang, J. Bi, H. Zhang, L. Zhang, A framework to quantify the pitfalls of using traceroute in AS-level topology measurement, IEEE JSAC 29 (2011) 1822–1836.

[52] K. Chen, D. R. Choffnes, R. Potharaju, Y. Chen, F. E. Bustamante, D. Pei, Y. Zhao, Where the sidewalk ends: Extending the Internet AS graph using traceroutes from p2p users, in: Proc. ACM CoNEXT, 2009.

[53] P. Gill, M. Schapira, S. Goldberg, Let the market drive deployment: A strategy for transitioning to BGP security, in: Proc. ACM SIGCOMM, 2011.

[54] R. Lychev, S. Goldberg, M. Schapira, BGP security in partial deployment: Is the juice worth the squeeze?, in: Proc. ACM SIGCOMM, 2013.

[55] M. Suchara, I. Avramopoulos, J. Rexford, Securing BGP incrementally, in: Proc. ACM CoNEXT, 2007.