Stony Brook University

Title of Thesis

A Thesis Presented

by

Muhammad Ruwaifa Anwar

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Master of Science

in

Computer Science

Stony Brook University

December 2016

Stony Brook University

The Graduate School

Muhammad Ruwaifa Anwar

We, the thesis committee for the above candidate for the

Master of Science degree, hereby recommend

acceptance of this thesis.

Phillipa Gill – Thesis Advisor
Research Assistant Professor, Computer Science Department

Samir Das – Thesis Committee Member
Professor, Computer Science Department

Michalis Polychronakis – Thesis Committee Member
Assistant Professor, Computer Science Department

This thesis is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Thesis

Investigating Interdomain Routing in the Wild

by

Muhammad Ruwaifa Anwar

Master of Science

in

Computer Science

Stony Brook University

2016

Models of Internet routing are critical for studies of Internet security, reliability and evolution, which often rely on simulations of the Internet's routing system. Accurate models are difficult to build and suffer from a dearth of ground truth data, as ISPs often treat their connectivity and routing policies as trade secrets. In this environment, researchers rely on a number of simplifying assumptions and models proposed over a decade ago, which are widely criticized for their inability to capture routing policies employed in practice. This thesis makes the following two contributions:

- Investigating Interdomain Routing Policies.

  First we put Internet topologies and models under the microscope to understand where they fail to capture real routing behavior. We measure data plane paths from thousands of vantage points, located in eyeball networks around the globe, and find that between 14-35% of routing decisions are not explained by existing models. We then investigate these cases, and identify root causes such as selective prefix announcement, misclassification of undersea cables, and geographic constraints. Our work highlights the need for models that address such cases, and motivates the need for further investigation of evolving Internet connectivity.

- Detecting BGP hijacks and interceptions

  We develop a system to detect BGP hijacks and interceptions in near real-time. When BGP was designed, the security challenges were not kept in mind. BGP lacks techniques like path validation and origin verification, as a result malicious ASes can advertises prefixes they do not own and can redirect the traffic to themselves. This is called BGP hijacking. Similarly, malicious ASes can partake man in the middle attack by routing traffic to the legitimate owner of the prefixes after redirecting first to themselves. This type of attack is called man in the middle attack. We develop a system to observe BGP announcements and updates in real time. We use combination of heuristics based on control plane and data plane (targeted traceroutes data) to separate malicious BGP announcements from legitimate announcements

Table of Contents

List of Figures

List of Tables

Measuring Interdomain Routing Policies

Models of Internet routing are critical for studies of Internet security, reliability and evolution, which often rely on simulations of the Internet's routing system. Accurate models are difficult to build and suffer from a dearth of ground truth data, as ISPs often treat their connectivity and routing policies as trade secrets. In this environment, researchers rely on a number of simplifying assumptions and models proposed over a decade ago, which are widely criticized for their inability to capture routing policies employed in practice. In this study we put Internet topologies and models under the microscope to understand where they fail to capture real routing behavior. We measure data plane paths from thousands of vantage points, located in eyeball networks around the globe, and find that between 14-35% of routing decisions are not explained by existing models. We then investigate these cases, and identify root causes such as selective prefix announcement, misclassification of undersea cables, and geographic constraints. Our work highlights the need for models that address such cases, and motivates the need for further investigation of evolving Internet connectivity

Research on existing and new protocols on the Internet is challenging because key aspects of the network topology are hidden from public view by interdomain routing protocols. Further, deploying new protocols at Internet scale requires convincing large numbers of autonomous networks to participate. As a result, networking researchers rely on assumptions, models, and simulations to evaluate new protocols [13, 26], network reliability [20, 41], and security [1, 16, 24]. Our existing models of interdomain routing [11], however, have important limitations. They are built and validated on the same incomplete topology datasets, typically routes observed via route monitors such as RouteViews and RIS [33, 39]. These vantage points expose a large fraction of paths from global research & education networks (GREN) and core networks, but they are incomplete in two keys ways. First, they expose few paths to and from eyeball and content networks. Second, they do not expose less preferred paths that would be used if the most preferred path was not available. As a result, they do not capture partial

peering, more complex routing policies based on traffic engineering, or load balancing and the rich peering mesh which exists near the edge of the network [35]. While limitations of our existing models are well known [27, 29, 35]–and are even being addressed in recent work [15]–we lack a solid understanding of how much these limitations impact our ability to accurately model the interdomain routing system. Recent work has attempted to address this issue by observing destination based routing violations in control plane data [28] and by surveying a population of network operators about their policies [12]. However, these approaches are limited in terms of scale and their ability to observe behavior at the network edge. In this paper, we take a systematic approach to understand how our models of routing policies [11] hold in practice. We leverage a combination of data plane measurements covering the network edge (Section 3.1) and control plane experiments which allow us to directly measure relative preference of routes (Section 3.2). We create a methodology that accounts for numerous potential causes of violations to our assumptions including sibling ASes [4], complex AS relationships [15], prefix-specific routing policies, and the impact of geography. We investigate the prevalence of each of these causes in AS-level paths observed via measurements of the data and control planes. We revisit generally held assumptions and models of Internet routing. Our goal is not to measure a complete Internet topology; rather, we seek to improve our understanding of routing decisions made by ASes when routing their traffic.

Towards this goal we make the following observations for our measured paths:

- Known hybrid and partial transit relationships (e.g., those explored in [15]) contribute a surprisingly small amount to unexpected routing decisions.

- Per-prefix routing policies appear to explain 10-20% of unexpected routing decisions, where an AS chooses a longer or more expensive path than our model predicts.
- We find that some large content providers like Akamai and Net- flix are destinations for a large fraction of unexpected routing decisions (21% and 17%, respectively).
- Routing decisions vary based on geography. We find that paths traversing multiple continents deviate from our models more, owing to undersea cable ASes which are not accounted for in our models. We also observed a tendency for ASes to prefer non-international paths when endpoints are in the same country.

Our results highlight areas where more investigation would yield the largest payoff in terms of improving our accuracy when modeling AS relationships and routing policies. We also identify key areas, specifically investigating prefix-specific routing policies, where additional vantage points and looking glass servers could improve the fidelity of our AS topology data.

Modeling Interdomain Routing

The now standard model of routing policies was developed by Gao and Rexford [10, 11] based on seminal work by Griffin, Sheppard, and Wilfong [17] and Huston [18, 19]. In this model, ASes connect to each other based on business relationships:
(1) customer-provider, where the customer pays the provider, and (2) peer-to-peer, where the ASes exchange traffic at no cost. This model gives the following view of local preferences and export policies, based on the economic considerations of ASes:

Local Preferences. An AS will prefer routes through a neighboring customer, then routes through a neighboring peer, and then routes through a provider. In other words, an AS will prefer cheaper routes.

Export Policy. A customer route may be exported to all neighboring ASes. A peer or provider route may only be exported to customers.

This model is sometimes augmented with the assumption that ASes only consider the next hop AS on the path when making their routing decisions. This simplifies analysis and makes debugging more tractable [20]. Simulation studies also often restrict path selection to the shortest among all paths satisfying Local Preference and use tie-breakers to induce unique routing decisions when AS path lengths are same [13, 14].

While the above model and variations thereof have been used in many studies (e.g., [1, 13, 16, 21, 41]), it is well known that this model fails to capture many aspects of the interdomain routing system [27, 29, 35]. These aspects include AS relationships that vary based on the geographic region [15] or destination prefix, and traf fic engineering via hot-potato routing or load balancing. Prior work has used traceroute measurements and BGP data to address some of these issues (e.g., [27, 29]); however, these measurements only offer a glimpse into ASes' routing preferences. Namely, they expose only the set of paths that are in use at the time of measurements. In contrast, we use active control plane experiments (PEERING [37]) to expose less preferred paths. Further, these datasets have poor or no coverage of paths used by edge networks [7]. On a smaller scale, network operators were surveyed about their routing policies to better understand how our models correspond to practice [12], but the scale and representativeness of a survey approach makes generalizing these observations infeasible.

Methodology

We aim to understand the gap between interdomain routing models and empirically observed behavior on the Internet. Our methodology combines two measurement techniques to gain better visibility into interdomain routing policies. First, we passively observe routing decisions on paths towards popular content networks (Section 3.1). We leverage the RIPE Atlas platform which provides a large collection of vantage points located around the world for our traceroute measurements. We thus observe routing decisions for broad range of hosts from variety of vantage points. One limitation of this approach lies in its passiveness as it only provides information about paths that are in use at the time of measurements. We do not get any information about the alternate paths available to an AS. Our second technique (Section 3.2) overcomes the above mentioned limitation and exposes less preferred paths for different ASes. We use PEERING [2, 37, 40] to selectively poison BGP announcements and force ASes to choose an alternate path, then we use RIPE Atlas probes as vantage points to run traceroutes towards poisoned prefixes to observe these alternate paths. This approach of actively probing routing decisions enables us to discover less preferred paths and also reverse engineer the BGP decision process. However, the PEERING platform is currently limited to few locations from which we can send poisoned announcements.

Passively observing route decisions

It is well known that a disproportionately large amount of Internet traffic originates from a few popular content providers [23, 36]. However, there is little empirical data about the paths this traffic takes [23]. We target these paths with our measurements. Note that it is not our goal to observe routing decisions for the entire Internet. Rather, we focus on the more tractable task of measuring a subset of important Internet paths (those carrying most traffic) from a diverse set of vantage points, and putting those paths under the microscope to understand how and why they differ from paths predicted by routing models.

Selecting content providers

We consider a list of the top applications from Sandvine [36] and top Web sites from Quantcast [31]. From these lists, we isolate top HTTP and non-HTTP hosts in terms of number of downstream bytes and number of visits. Finally, we arrive at a list of 34 DNS names representing 14 large content providers.

Vantage Points

RIPE Atlas has broad global coverage, but is known to have a disproportionate fraction of probes skewed towards Europe.

AS type	Probes	Distinct ASes	Distinct
Stub-AS	787	333	106
Small ISP	581	188	78
Large ISP	56	109	51
Tier 1	69	8	3

Table 1: Distribution of selected RIPE Atlas probes

To avoid a bias towards European ASes, we picked equal number of probes from each continent. For every continent, we picked probes in a round robin fashion from different countries and ASes so that selected probes cover a wide range of ASes. Table [PUTNUMBER] summarizes the location of these probes in terms of AS type using the categorization method of Oliveira et al. [30]. The bulk of the probes are located near the

network edge in stub and small ISP networks. To measure paths to content providers, each RIPE Atlas node performs a DNS lookup for each of the 34 content DNS names, and then performs a traceroute to the resolved IP. We use 1,998 RIPE Atlas probes located in 633 ASes, distributed according to our sampling methodology. Data set. We used maximum probing rate allowed by RIPE Atlas to perform 28,051 traceroutes towards selected hosts. These traceroutes ended up in a total of 218 destination ASes. The number of destination ASes is large relative the number of content providers because large numbers of content servers are hosted outside the provider's network (e.g., inside ISPs) [5]. We convert the traceroute-based IP-level paths into AS paths using the method described by Chen et al. [7]. Since interdomain routing is destination based, we can observe routing decisions for all ASes along the path to a given destination. We thus observe routing decisions for a total of 746 ASes.

Actively probing route decisions

Passive measurements observe only the most preferred route for an AS toward a destination. We use PEERING [2, 37, 40] to expose alternate, less preferred routes and to attempt to reverse engineer BGP decisions.

PEERING operates an ASN and owns IP address space that we can announce via several upstream providers. PEERING allows us to manipulate BGP announcements of its IP prefixes and observe how ASes on the path react. We used PEERING to announce prefixes using six US universities (Georgia Tech, Clemson, University of Southern California, Northeastern, Stony Brook, and Cornell) and one Brazilian university as providers. We change announcements at most once per 90 minutes to allow for route convergence and avoid route flap dampening. We use prefixes allocated to the PEERING research testbed reserved for our experiments; these prefixes carry no real traffic beyond our measurements.

Discovering alternate routes

We start announcing an IP prefix from all PEERING locations in an anycast announcement. At each round, we observe the preferred route at a target AS T and the next-hop neighbor N that T is using to route toward our prefix. We then poison N, i.e., add N's AS number to the path [3, 9], to trigger BGP loop prevention at N and cause N to no longer have a path to our prefix (and stop announcing a route to T). This forces T to choose a different route, through a different neighbor N 0 . We repeat this process in consecutive rounds, poisoning the newly-discovered neighbor, to identify all neighbors and routes T can use toward our prefixes. When we observe different routes at the target AS T (through different neighbors) from multiple vantage points (e.g., due to

different routing preferences at different geographic locations), we run the algorithm for each vantage point separately. We can potentially execute this algorithm for each AS in the topology as the target AS. A similar experiment was performed by Colitti [9]; here, we use the same mechanism with a more diverse set of providers and with a different goal.

We insert all poisoned ASes into a single AS-set, and surround the poisoned AS-set with PEERING's AS number. This limits ASpath length, prevents inference of non-existent inter-AS links, and allows operators to identify the poisoning.

Reverse engineering BGP decisions

In addition to the experiment to discover alternate routes, we conduct a complementary experiment to infer BGP decision triggers. We first announce an IP prefix from one PEERING location (called the magnet), wait five minutes to allow for route convergence, then announce (anycast) the same IP prefix from all other PEERING locations. After we anycast the prefix, an AS may change to a new route with higher LocalPref, shorter AS-path length, or better intradomain tie-breakers, as specified in the BGP decision process [8]. If an AS x keeps using the route toward the magnet after we anycast the prefix, we check if the magnet route is cheaper according to the Gao-Rexford model or has shorter AS-path length than all other routes we observed from x. If none of these checks are satisfied, we infer AS x is using intradomain costs or route age (the last tie-breaker before router ID) as a tie-breaker. If AS x did not choose the route to the magnet, we check if the chosen route is cheaper or shorter than the route to the magnet. If none of these checks are satisfied, we infer AS x is using intradomain costs as a tie-breaker. We repeat this process using each PEERING location as the magnet. We also check whether the route chosen after we anycast the prefix is more expensive according to the Gao-Rexford model or is the same cost but has longer AS-path length than other routes we observed, which is a violation of the model. The route to the magnet may become unavailable when a downstream AS receives and chooses a more preferred route; in these cases we consider the downstream AS's decision.

Vantage points (VPs)

We perform traceroutes from 96 RIPE Atlas probes and approximately 200 PlanetLab nodes every 20 minutes, and collect BGP feeds every 15 minutes from RouteViews and RIPE RIS to monitor paths toward PEERING prefixes. We use the maximum number of RIPE Atlas probes allowed within daily probing budget limits. We implement a greedy heuristic that picks probes to maximize the number of ASes traversed on the default paths toward PEERING locations.

Data set

We needed a total of 188 distinct poisoned announcements to infer preferences for all 360 target ASes we observe on paths toward PEERING (some poisonings are useful for multiple target ASes). We observe 739 inter-AS links. We find 45 inter-AS links that are not in CAIDA's AS-relationship database, 10 of which (22.2%) can only be observed with poisoned announcements.

Comparison with existing models

We compare paths observed in our passive and active measurements with CAIDA's topology of inferred inter-AS relationships. We aggregate five topologies (Oct. 14 to Feb. 15) inferred using the method presented by Luckie et al. [25]. We aggregate these snapshots to mitigate the impact of transient link failures on the topology used in our analysis. When inferences conflicted, we took the majority poll of inferred relationships while assigning higher weight to more recent inferences, i.e., if the latest two months had the same inference, we used that inference regardless of the first three months. We use this topology to compute all paths that satisfy the Gao-Rexford (GR) model described in Section 2.

We compare the measured paths with all paths satisfying the GR model computed using CAIDA's inferred relationships. We consider two properties: (1) whether the measured path satisfies the GR model of local preference, and (2) whether the measured path has the same length as the shortest paths satisfying the GR model of local preference. Based on this we classify routing relationships as either obeying GR local preference; i.e., using the neighbor with the Best Relationship type **(Best)**, routing based on shortest path **(Short)**, or a combination of the two.

For our active probing measurements, we consider the order in which the target AS T chooses paths. Again, we consider two properties: (1) whether the relationship between T and the next-hop on the first path is equal or better than the relationship with the nexthop on the second path, and (2) whether the first path is shorter or equal in length as the second path. We similarly label the observed decisions which obey property (1) as Best, and those that obey (2) as Short. We have limited visibility on what path the second neighbor exported to T when T chose the first path. When labeling decisions, we assume the second neighbor exported the second path to T when T chose the first path. We verified this assumption holds for the results we report.

In both cases, the sets should be treated as disjoint, with ASes that obey both Best and Short path policies appearing only in the **Best/Short** category. Observations which follow neither of these properties are considered inconsistent with existing models (i.e., **NonBest/Long** category). There can be, however, some cases when a path suggested

by CAIDA's inferences might not exist in practice. One of the reasons can be incomplete or erroneous inferences in the topologies. In addition, an AS might apply more complex filters than suggested by Gao-Rexford model when deciding which paths to advertise to neighbors (Section 4.3 discusses this in more detail).

How often to the model hold

We now consider how empirically observed AS paths compare with those predicted by GR model. We then investigate how often deviations can be explained by known sources of inaccuracies.

Encouragingly, we find that a majority of routing decisions (64.7%) for passively observed measurements are correctly inferred by the commonly used GR model; however, a significant fraction (34.3%) do not follow that model. Figure 1 (Simple) characterizes the observed routing decisions based on whether the path chosen is Best or Short. We find only a small number of cases (8.3%) where decisions can neither be explained by Best nor by Short path selection. In the following sections, we explore the reasons behind these decisions that differ from model-based predictions.

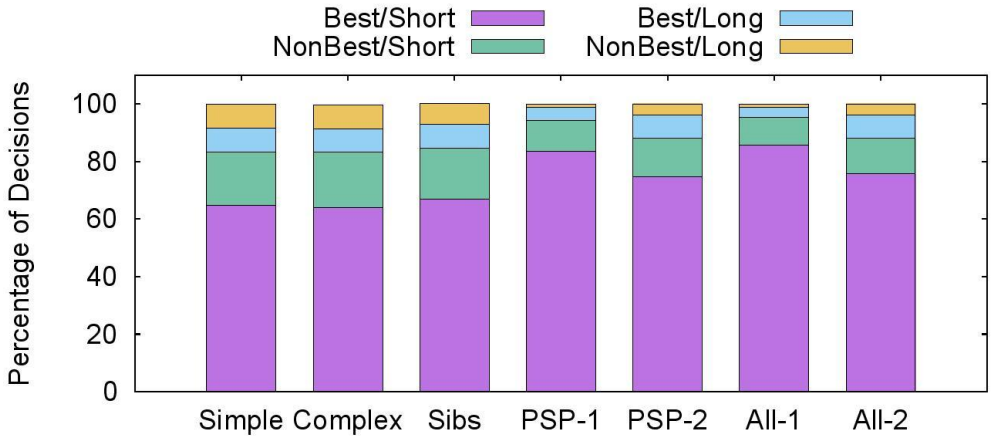

Figure 1: Breakdown of routing decisions observed by taking into account complex relationships (Complex), siblings (Sibs), prefix-specific policies (PSP-1, PSP-2) and by combining complex and siblings relationships with both criteria of prefix-specific

Complex routing relationships

A well known limitation of existing routing policy models is the simplification of relationships into either customer-provider or settlement-free peering relationships. Recent work by Giotsas et al. addresses this limitation by augmenting relationship

inferences with cases of hybrid relationships (i.e., ASes whose arrangements vary based on location) and partial transit relationships (i.e., ASes who will behave as providers, but only for a subset of prefixes) [15]. The hybrid relationship dataset contains pairs of ASes and the corresponding cities where relationships differ for a given AS pair. To use this dataset, we use the geolocation data from [6], which offers good coverage of infrastructure IPs such as routers. For each pair of ASes in each AS path, we geolocate corresponding IP addresses and if the geolocation data points to the same city as mentioned in hybrid relationship dataset for that AS pair, we use the hybrid relationship. Figure 1 (Complex) shows the breakdown of routing decisions observed taking into account these complex relationships. Interestingly, we find that taking these relationships into account has nearly no impact on the classification in our dataset (less than 1% change).

Sibling ASes

The mapping between AS numbers and organizations is not oneto-one [4]. Many organizations manage multiple AS numbers, either for geographic regions (e.g., Verizon with ASNs 701, 702, and 703) or due to mergers (e.g., Level 3 (AS 3356) and Global Crossing (AS 3549)).

Cai et al. [4] present a technique to map organizations to ASes by using attributes like organization IDs, email addresses and phone numbers found in whois information of ASes. We take a similar approach to identify AS siblings, but our approach differs in two key ways. First, we focus only on e-mail addresses in whois data, which previous work identified as the field with best precision and recall [4]. Second, we use DNS SOA records to identify different e-mail domains that belong to the same organization. For example, dish.com and dishaccess.tv share the dishnetwork.com authoritative domain. We also remove groups where the e-mail address is hosted by a popular e-mail provider (e.g., hotmail.com), or regional Internet registry (e.g., ripe.net). This results in a total of 94 sibling groups identified in our traceroute data set.

For every non GR decision that an AS makes, we check whether the AS chooses a path via a sibling. If the path is via a sibling, we mark this decision as satisfying the Best condition. Figure 1 (Sibs) shows the result of this change—3.9% more decisions are classified as Best/Short.

Prefix-specific policies

Interdomain routing is often abstracted to the level of a destination AS. However, in practice routing is based on IP prefixes which may be subject to different export policies by their originating AS (e.g., forwarding prefixes hosting enterprise-class services to a more expensive provider). While Giotsas et al. consider partial transit [15], which is a type of prefix-specific policy, they do not explicitly consider per-prefix policies as implemented by origin ASes.

We use two criteria to identify prefix-specific policies based on correlation with BGP data obtained from Routeviews/RIPE [34, 39]. Given an origin AS (O), a neighbor N and a prefix P: Criteria 1 do not assume the edge N − O exists for prefix P unless we observe O announcing P to N in the BGP data. Criteria 2 is similar to Criteria 1, except that we require that we observe at least one prefix announced from O to N before applying Criteria 1. The first criteria can be seen as being more aggressive whereas the second aims to ensure that our observation is actually caused by selective prefix announcement and not poor visibility.

Figure 1 (PSP-1, PSP-2) shows the breakdown of routing decisions using Criteria 1 and 2 above, respectively. We find that prefixs-pecific policies account for a significant fraction (10-19%) of unexpected routing decisions. Combining Criteria-1 and Criteria-2 separately with simple, complex and siblings relationships, yields 85.7% and 75.7% of decisions for Best/Short category respectively (Figure 2, All-1, All-2). One limitation of these approaches is that we only check prefix-specific policies for origin ASes. Other limitation is incomplete visibility in BGP control plane data.

Validation In order to validate the cases of prefix-specific policies, we try to find a Looking Glass server hosted by the neighboring AS of the AS originating the prefix being examined. There were a total of 630 cases of prefix-specific policies involving 149 unique neighboring ASes. We were able to find looking glass servers in 28 of the neighboring ASes. Using these looking glass servers we manually verify 100 cases of prefix-specific policies and confirm that applying Criteria 1 was correct 78% of the time.

Active BGP Measurements

Using our active BGP measurements, we discover alternate routes. We study whether the sequence of alternate route choices match existing models and infer which step of the BGP decision process led to each route. We report results for experiments performed between Feb. 25th and Apr 27th, 2015. Alternate routes. We analyze AS

routing choices when we use PEERING to discover alternate, less preferred routes. We compare the sequence of routes chosen by target ASes with

BGP decision	Bgp feeds	Traceroutes
Best Relationship	435	228
Shortest Path	155	158
Intradomain Tie breaker	155	84
Oldest route (magnet)	24	9
Violation	179	58
Total	945	537

Table 2: BGP decisions observed after we anycast a prefix previously announced from a single (magnet) location.

AS-relationships database. Out of the 360 ASes we targeted, 310 (86.1%) chose routes following both Best and Shortest (as defined in Sec. 3.3); 29 (8.0%) chose routes following Best only; 18 (5.0%) following Shortest only; and 3 (0.8%) did not follow either property. We discuss the three observations that did not satisfy either property to illustrate limitations of current models.

One violation occurs for a European network E that routes via Open-Peering (AS20562)–a transit relationship identified from RPLS entries in public routing databases. After poisoning OpenPeering, E routes through (a likely peer-to-peer relationship) with AMPATH (AS20080) at AMS-IX. We list this as a violation because CAIDA identifies OpenPeering as a provider for E and AMPATH as a peer of E. Interestingly, the second route is the suffix of the first route (i.e., the route through OpenPeering also reaches PEERING through AMPATH at AMS-IX), indicating the first route includes an unnecessary detour. Relationships are complex; transit and peering relationships may be preferred one over the other. Models with finer granularity for ranking neighbors of an AS may resolve these issues [27].

Another violation occurs at a US university U. The university first routes through Internet2 (AS11537) toward one of the PEERING locations in the US. After we poison Internet2, U routes through AMPATH (AS20080) toward the PEERING location in

Brazil. We list this as a violation because CAIDA identifies Internet2 as a provider and AMPATH as a settlement-free peer of U. Our last observed violation is similar, where a European network first routes through Switch (AS559, identified as a provider) and then routes through NCSA (AS10764, identified as a settlementfree peer) to reach PEERING after we poison Switch. These violations indicate that identifying links used as back-up might improve our routing models.

Reverse engineering BGP decisions

We now turn to our second control plane experiment, where we use anycast to explore considerations such as route age on routing decisions. Table 2 shows the root cause behind BGP routing decisions. Although most decisions are made based on relationship and path length, more than 17% of decisions are made based on intradomain tie-breakers and route age, which are not considered in and could improve current models.

Limitations

BGP poisoning does not work when BGP loop prevention is disabled or when ASes filter poisoned announcements [20, 22]. Intermediate ASes between PEERING locations and target ASes may prevent us from controlling routes exported to the target AS. These factors limit our ability to identify all routes available to and neighbors of target ASes. We consider the subset of routes we observed and neighbors we identified. Moreover, our results for these experiments cover a small fraction of the Internet and are probably biased toward academic and research networks. Our control plane techniques, however, are general and could be used by other networks to cover different portions of the Internet. We believe better coverage and visibility would result in discovering more violations. To this end, we are working to extend the PEERING platform and RIPE has configured periodic measurements from a diverse set of Probes toward all PEERING prefixes.

Skewness by source and destination

We now investigate which source and destination ASes account for most of the routing decisions which deviate from our model. Figure 2 (a) and (b) shows the cumulative fraction of routing decisions which violate either the Best or Short condition (i.e., the AS chooses a path that is longer or more expensive than we would expect). If violations were evenly distributed across ASes, the curves would fix y = x; otherwise, some ASes are responsible for a disproportionately larger (or smaller) fraction of violations. We find

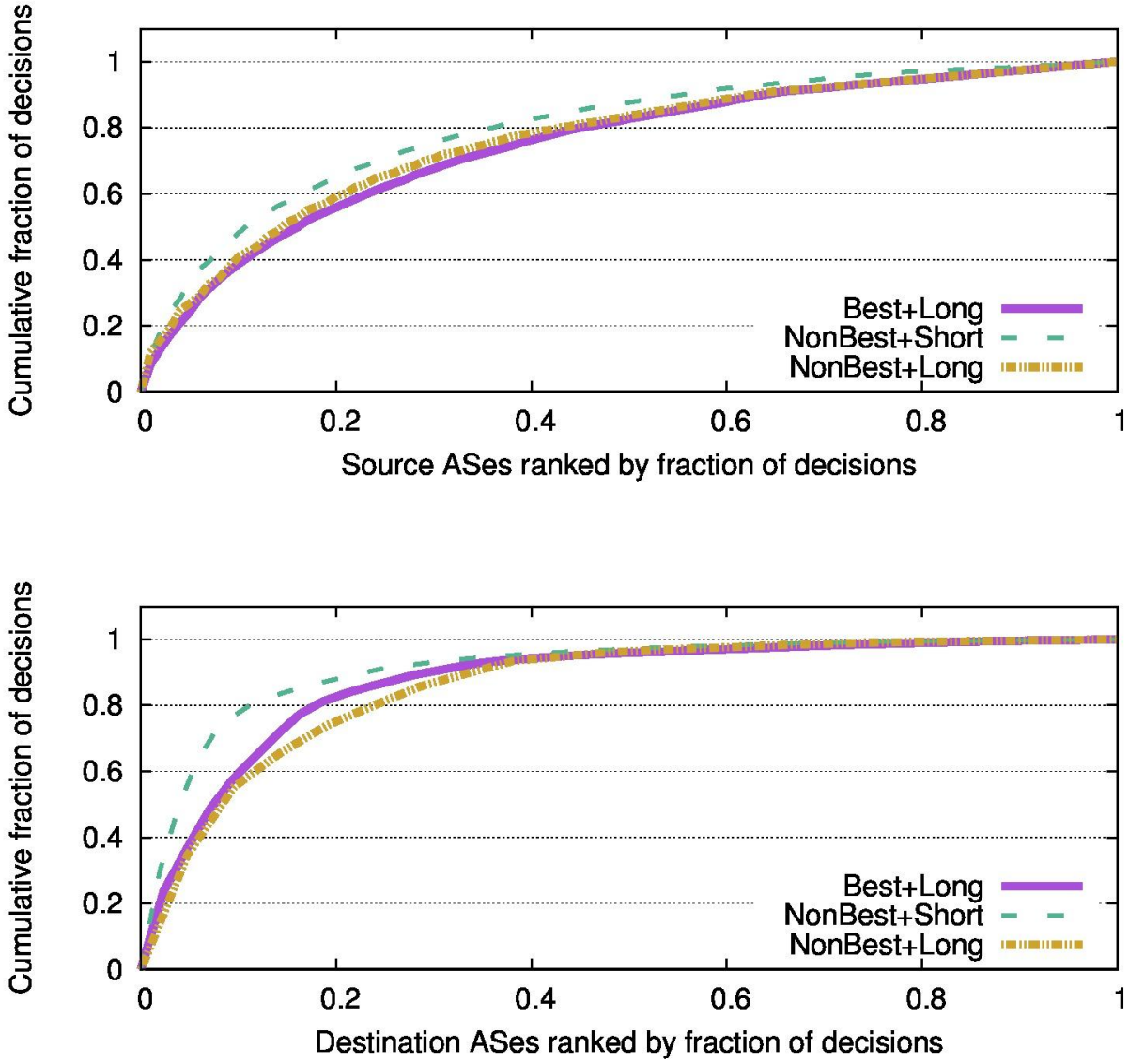

Figure 2: CDF plot of the fraction of violations (x-axis) explained by source and destinations ASes (y-axis). Violations observed in our dataset are skewed significantly toward Akamai and Netflix (21% and 17% of total NonBest/Short violations respectively). The skew for source ASes is less prominent.

this effect is present in both plots, but more prominently for destination ASes. We focus on the latter.

Destination ASes owned by Akamai account for 21% of violations. Of these, Cogent (AS174) is the most common source, responsible for 3.4% of these Akamai related violations. These Cogent-Akamai violations tend to occur when Cogent prefers a peer-

to-peer path through a Tier-1 AS over a longer customer route towards Akamai. Netflix's AS is the destination on 17% of paths with violations. Of these, nearly a quarter (24%) are due to a stale inter-AS link in CAIDA's topology, which included a direct link between AS3549 and Netflix that no longer exists according to RIPE ASN Neighbor History [32]. For source ASes, the distribution is less skewed. Cogent and Time Warner are the top two sources, responsible for 4.1% and 2.2% of violations, respectively. The skew for source ASes is less prominent.

Impact of Geography

We next consider the role of geography on routing decisions. First, we isolate traceroutes that stay within a continent (Continental traceroutes), i.e., all hops stay inside a given continent based on geolocating router IP addresses. Figure 3 shows the breakdown in decisions in the continental traceroutes (45% of our dataset). The fraction of decisions explained by GR for continental traceroutes is significantly greater than for intercontinental ones.

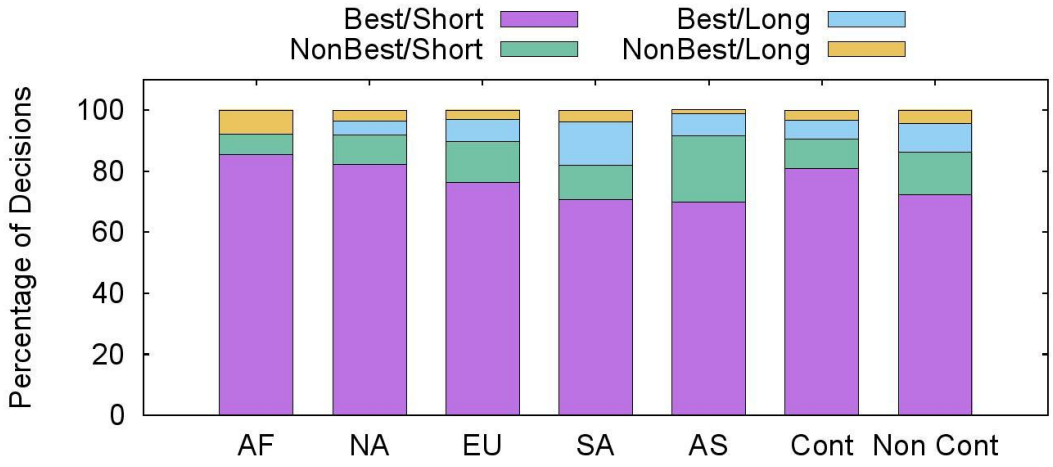

Figure 3: Breakdown of routing decisions for traceroutes that stay within continents of Africa (AF), North America (NA), Europe (EU), South America (SA), Asia (AS) and all continents combined (Cont), and for intercontinental traceroutes (Non Cont).

Domestic Paths

Next we focused on traceroutes where we infer that the entire traceroute stayed within a single country, but there is a better multinational Best/Short path (in the CAIDA data), which we define to be a path with at least one AS registered (via whois data) in a

country outside the source and destination AS's country. We find that more than 40% of non-Best/Short decisions can be explained by avoiding alternative multinational paths. One limitation of this approach is that even for the ASes that reside in multiple countries, whois data still points to just one country or when an AS spans across multiple regional internet registries then each RIR shows different country as the origin of that AS. Table 3 details the non-Best/Short decisions explained by ASes preferring domestic routes.

Undersea cables

Undersea cable ASes are a critical component of Internet topologies that previous works overlook [15, 25]. While some cables are jointly owned by large ISPs, e.g., Pan-American Crossing, Americas-II (owned by AT&T, Sprint, and many others), we observed that others, e.g., EAC- C2C (PACNET), are operated by independent organizations using their own allocated ASNs and IP prefixes. Because these cable operators only provide pointto-point transit along the cables (i.e., they do not originate traffic and peer in locations proportional to cable landings), they resemble high-latency, high-cost IXPs and thus confuse existing AS relationship models. As such, we need techniques to identify cable ASes and correct their relationships in inferred topologies.

Continent	**Non Best/Short Decisions**
Asia	40.1%
Africa	62.5%
Europe	64.3%
North America	10.9%
Oceania	62.9%
South America	66.6%

Table 3: Continent wise violations

We use a list of undersea cables from the TeleGeography Submarine Cable Map [38] to identify ASes for undersea cable operators. Overall, cable-ASes appear on less than 2% of paths but most of the decisions (51.2%) involving cable-ASes caused deviations from Best/Short paths. Table 4 shows fraction of each type of decision explained by undersea cable ASes. Violation type Pct. of decisions explained Non-Best & Short 3.0%

Best & Long 6.5% Non-Best & Long 4.5% Table 4: Fraction of decisions of each type that can be attributed to undersea cables.

Conclusion

In this work, we investigated how interdomain paths predicted by state-of-the-art routing models differ from empirically observed routes. We found that while a majority of paths in our dataset agree with models, more than a third do not. We explained a significant fraction of these differences due to factors such as sibling ASes, selective prefix announcements and undersea cables. Further, we investigated how the models hold up when comparing with groundtruth routing preferences revealed using PEERING announcements, and identified AS behavior that is not included in existing models. As part of future work, we are continuing to investigate cases of routing decisions that violate today's models, and we aim to incorporate our findings into new models of Internet routing.

BGP Hijacks and Interception Detection

BGP was designed when the Internet was comprised of a few ASes, hence security was not in the top of the mind. The protocol doesn't provide mechanisms to validate or authenticate the messages being sent by ASes, which makes it is highly vulnerable to routing incidents caused by misconfiguration or attacks [42,43,44], including hijacking. These incidents include large-scale route leaks [46], where an AS originates a large number of prefixes allocated to other ASes (e.g., the China Telecom incident [46]) and more suspicious forms of path manipulation, where an AS may announce a path that does not actually exist in the AS-graph [45]. Another type of BGP hijacks is the interception attack, the malicious AS gets the traffic destined to some other ASN, sniffs it and send it back to the unsuspecting AS

Related Work

A lot of past works have focused on detecting BGP hijacks. Initial works Cyclops [47] and PHAS [48] use information only from the control plane data and raise alarms based on anomalies found in it. These systems, however, suffer from the large number of false positives. To tackle this problem, PHAS and Cyclops have subscription based system in which the prefix owners are notified if their prefix is originated by other autonomous systems. Though high on false positives, the solutions provided by these works are easily deployable.

Some works looked at only dataplane based anomalies to detect hijacks. These systems issue periodic traceroutes towards the prefixes and observe changes in the AS paths[49,50]. These solutions offer less false positives than control-plane-only based systems, however these systems suffer from scaling issues as it is infeasible to monitor all the prefixes all the time and also these systems require strategic placement of vantage points which is not possible with limited number of Planet Lab nodes used in these studies.

Some works combined both data plane and control plane information. Initial work in this domain was presented by Xu. et al [51]. On the detection of control plane anomalies, the system carries out dataplane measurements from the Planet Lab vantage points with modified software which do host based fingerprinting analysis to determine the

cause of the anomaly. This solution was indeed scalable as they only monitored BGP updates and the dataplane measurements were spawned only in case of control plane anomaly. However, the host fingerprinting required looking at TCP timestamps, IP IDs etc which required customized software on the Planet Lab nodes, thus making the system harder to deploy. Another similar system, Argus[52] also used both control plane and dataplane information from route servers or looking glasses. Similar to [51], they issued dataplane measurements (i.e pings) only in case of control plane anomalies. They studied reachability of the prefixes both in terms of control plane and dataplane and used a statistical approach to classify the events as hijacks or benign ones. The system laid too much emphasis on using looking glass servers and carrying out dataplane measurements from those looking glass servers. These servers are limited and not all such servers provide means to carry out dataplane measurements. Thus too much reliance on external hardware poses some deployment challenges. Since the reachability in terms of dataplane, meant classifying those events as benign, so this system could not detect BGP interceptions.

Detecting BGP interceptions was infact ignored in earlier works too. A part of reason behind this lack of focus might be due to scant BGP interception examples from the real life. Infact, before Renesys presented a real life example of BGP interception [9], the most attention BGP interceptions ever got was the conceptual demonstration by Psilov and Kapella at Defcon'06[52]

Apart from above mentioned systems, [54] identified BGP hijacks from a large dataset containing known events of spam. [53] also found spam campaigns using BGP hijacks. These works however fell short in presenting a system to detect future BGP hijacks.

Our system employs both control plane and data plane measurements to detect BGP hijacks as well as BGP interceptions in near real time. We don't rely on looking-glass servers for dataplane measurements nor do we require any software modification in vantage points. We leverage vast data plane measurement infrastructures of RIPE Atlas and CAIDA to carry out traceroutes in the case of control plane anomalies. This makes our system easily deployable. Our methodology does not use rely solely on reachability so we detect BGP interceptions.

	Uses Control plane	Uses Data plane	False positive rate	Detects Interceptions	Deployment	Scalability
Cyclops	Yes	No	High	No	Easy	Poor
PHAS	yes	No	High	No	Easy	Poor
iSpy	No	Yes	Medium	No	Easy	Poor
Xu et.al	Yes	Yes	Low	No	Hard	Medium
Argus	Yes	Yes	Low	No	Medium	Medium
Our	Yes	Yes	Low	Yes	Easy	Easy

Table 4: Comparison among different BGP anomaly systems

System Architecture

We have developed a BGP hijacks and interception detection system, BGPSTREAM.

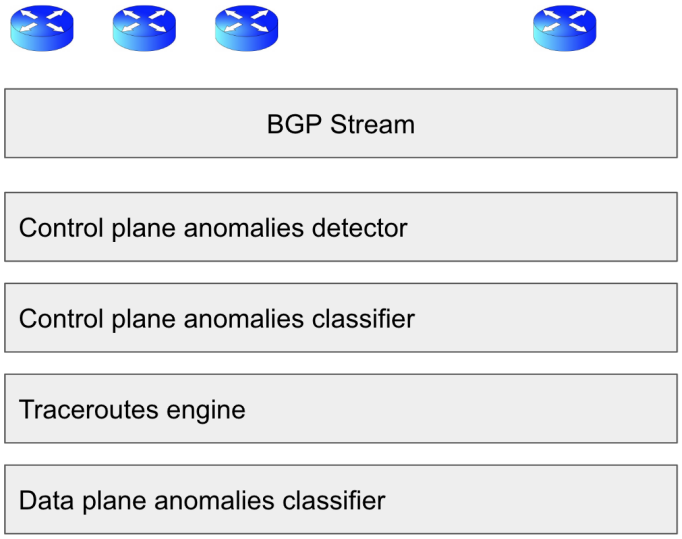

Figure 4: BGP Stream Architecture

The system start with getting BGP control plane feeds from multiple BGP route collectors from RIPE and RouteViews project. From this routing data, we look for different types of control plane anomalies. We then run different heuristics, to remove false positive from those anomalies. Then we move to carrying out targeted data plane

measurements (traceroutes). In response to suspicious control plane anomalies, we carry out traceroutes. Finally, we have our dataplane classifier, that tries to remove more false positives to get the list of highly suspicious events

We look for four different types of control planes anomalies:

1: MOAS (When a prefix is being advertised by multiple ASNs)
2: SubMOAS (When an ASN announces a prefix which belongs to the address space of another ASN)
3: New edges in AS graph (When a new AS edge is seen that's not seen before)

Not all of these events of these anomalies can be due to BGP hijacks or interceptions, we devise methodologies to identify those events that are false positive or part of normal Internet routing system. These anomalies are further explained in section x

Diagnosing BGP Hijacks and Interceptions

For each event detected as suspicious, the module for each of the four anomalies described above generates relevant meta-data, including which type of anomaly was observed, potential victim prefixes and ASes, and relevant AS paths to measure. To remove false positives from control plane anomalies and minimize the set of prefixes to traceroute to we use following resources:

CAIDA's AS relationship:
We use CAIDA's AS relationship dataset which gives us information about business relationship different ASes [1]. This is being updated on monthly basis by CAIDA.

Siblings ASes datasets:
Siblings are those ASNs that belong to same organization. Many organizations manage multiple ASNs for geographical reasons or commercial mergers. We use CAIDA's AS-to-Organization dataset to generate list of siblings ASNs [2]. The dataset assigns Organization IDs to ASNs, two ASNs are assigned same organization IDs if they are considered belonging to same organization based on similarities in various whois information fields like email addresses, contact phone, organization name, administrative contact etc. We consider those ASNS as siblings which are assigned same organization IDs by this dataset.

Private ASNs:
These ASNs are not allocated to organization for public routing. Most common type of these ASNs are private use ASNs. They are used to testing purposes normally and are not meant to be exposed to the Internet. The use of private ASN is more frequent in private networks that will never communicate directly with the Internet. Most ISPs utilize route filters to reject routes that contain private ASNs. However, sometimes private ASNs are still in ASpath of globally reachable IP prefixes. Along with private ASNs, we also check for unallocated ASNs like ASN reserved for IANA use etc. IANA specifies which ASNs are reserved for private use. [3]

Furthermore, we also use IXP prefixes to filter out false positives

Control Plane Anomalies

MOAS

A MOAS (Multiple Origin AS) occurs when a prefix is originated by multiple origin ASes. A MOAS event is uniquely identified by the pair <pfx, origin_set>, where origin_set is the list of ASes announcing the prefix pfx. A prefix can be associated to one and only one MOAS event at a time.
A MOAS event starts when the prefix pfx starts being announced by a specific origin_set (different from the list of origin ASes announcing it at the previous timestamp).

- t - <pfx, origin_set(t)>
- t+1 - <pfx, origin_set(t+1)>

where origin_set(t) != origin_set(t+1) and size(origin_set(t+1)) > 1.
A MOAS event ends when the prefix pfx starts being announced by a different origin_set (different from the list of origin ASes announcing it at the previous timestamp), or it stops being announced.
A MOAS event is ongoing, if the prefix keeps being announced by the same origin_set as it was before, a MOAS event is recurring if it has been already seen within a moving window of one week. Not all such anomalies can be result of hijacks/interception, we can have legitimate cases of MOAS or submoas due to IP anycast, IP transfers, prefix de-aggregation etc.

We classify a MOAS event as a legitimate event when:

- An IXP prefix is involved in the MOAS event

- Private ASN is responsible for the MOAS event
- All ASNs in a MOAS group belong to sibling ASNs
- All origin ASNs for a MOAS prefix can form a customer-provider chain where each ASN is single homed (each ASN has only one provider)

As far as the effect of duration of an event on its suspiciousness is concerned, we are currently not taking the duration in account. While most of the hijacks are short lived and longer the duration an event exists (say MOAS event), the lesser would be probability of the event being a malicious one. But still some malicious events are long lived as much as 8 months as reported by Vervier et.al . Similar observation was made by Ramachandran et.al
Using the rules above, we will curate a list of MOAS violations that are a normal part of the Internet's routing system and thus do not require further analysis through active measurements.

SubMOAS

A SubMOAS (subprefix MOAS) occurs when a sub-prefix is originated by a set of ASes sub-origins that differs from the set of ASes that originates one of its super-prefixes (super-origins).
For example, consider the prefix 181.48.0.0/13 originated by AS14080. Later, AS10620 announces 181.52.148.0/22.

Since the new prefix lies in the address space of already originated prefix and both prefixes have different ASNs, we will consider this event as a SubMOAS event. The prefix 181.52.148.0/22 is sub-prefix and AS 10620 is the sub-origins, the prefix 181.48.0.0/13 is super-prefix and AS 14080 is super-origins.

A SubMOAS event is uniquely identified by the tuple <super-prefix, sub-prefix, super-origins, sub-origins>, where sub-prefix is a more specific of the super-prefix, super-origins is the list of ASes announcing the super-prefix, and sub-origins is the list of ASes announcing the sub-prefix. A <super-prefix, sub-prefix> pair can be associated to one and only one SubMOAS event at a time.

A SubMOAS event starts when the <super-prefix, sub-prefix> start being announced by two different origins' sets, i.e.

- t - <super-prefix, super-origins>
- t - <sub-prefix, sub-origins>

where super-origins(t) != sub-origins(t).

A SubMOAS event ends when the super-prefix or the sub-prefix start being announced by a different origins' set (compared to those observed at the previous timestamp), or if one of them stops being announced.

We classify a subMOAS event as a legitimate event when:

- An IXP prefix is involved in the SubMOAS event
- Private ASNs are responsible for the SubMOAS
- All origin ASNs from both sub-prefix and super-prefix form one sibling group
- All origin ASNs from both sub-prefix and super-prefix can form a customer-provider chain where each ASN is single homed (each ASN has only one provider)

New Edge

We monitor the set of edges observed in BGP announcements (similar to how Argus tracks pairs of ASes observed in BGP paths). A new edge event is considered finished when the new edge is no longer observed in the considered sliding window. While a lot of cases of new edges can be simply due to previously unseen backup links but new edges can be seen in case of route-leaks (cite Malaysia telecom example) or when a hijacker inserts a fake ASN to show it's adjacency towards a particular ASN.

We classify new edges as legitimate if:

- Both ASNs are siblings to each other
- One or both ASNs are private ASNs

For other new edges,data plane measurements will help us validate that they are not due to interception or hijacks, in which case we will add them to a list of previously seen edges. As failures happen and ASes explore backup paths, over time we should gain a more complete view of all edges we should expect to see.

Control Plane anomalies analysis

Data source

We use one of the RouteViews collectors (RouteViews-2) [3] as our source of BGP control-plane data. RouteViews-2 has 44 peers across different parts of the world [4], most of them being fullfeed peers. Having multi-hop and full-feed peers, if there is some

routing change somewhere in the world, RouteViews-2 should be able to see it in most of the cases. We use pyBGPstream to process historical data from MRT dumps from different peers of RouteViews-2. pyBGPstream collects BGP feeds (updates, withdrawals, RIBs) from multiple peers, sorts them and presents in chronological order. We could have used more collectors in addition to RouteViews-2 but adding more collectors resulted in more data being processed, of which a large fraction was repetitive as multi-hoped RouteViews-2 already covers a broad range of control-plane instances generated worldwide. We analyze control-plane data spanning from June, 1st 2014 to August, 1st 2014.

We generate a view as the global state of routing information. A view stores all the prefixes which are seen by any peer of the collector. A prefix is removed from the view only when all the peers which previously made announcements, send withdrawal message for that prefix. We update our view with RIBs of all the peers after every two days of BGP time to cater cases of incomplete or lost messages due to data corruption or a peer going down.

Computing Durations

A sub-MOAS event starts when a sub-prefix and a super-prefix are announced from different ASNs and it finishes when either of these prefixes is no longer advertised. The duration of a sub-MOAS event depends on which type of a prefix is removed from the view. In some cases, a super-prefix no longer advertised means all its sub-prefixes are no longer sub-MOASes.

Alternatively, we can have events, where a super-prefix is also a sub-MOAS prefix itself, so all its sub-prefixes still remain sub-MOAS prefixes but to a different super-prefix. Therefore, we deal with each case of prefix removal separately to find out when a sub-MOAS event is actually finished. We divide prefix removals in four different categories:

1. The prefix is a sub-MOAS itself and it has no sub-prefixes.
2. The prefix is a sub-MOAS itself and it has sub-prefixes.
3. The prefix is not a sub-MOAS but has sub-prefixes.
4. The prefix is not a sub-MOAS itself neither it has sub-prefixes.

In the first case, we consider the sub-MOAS event as finished and calculate the duration of the event. In the second case, the prefix itself is considered no longer a sub-MOAS prefix but all its sub-prefixes still remain sub-MOASes and their super-prefix is now the former super-prefix of the prefix that is just removed. In the third case, all the sub-prefixes are considered over and corresponding sub-MOAS durations are calculated. In the fourth case, we do not do anything as it does not deal with sub-MOASes in any way.

Filtering out

We have filtered out those sub-MOAS events which include private ASNs. Private ASNs are normally used by networks with a single provider. According to RFC [5], private ASNs are not meant to be exposed to global Internet. ASes tend to strip off private ASNs from the ASPATH.

Since Private ASNs are mostly used internally by organizations, it is unlikely that an attacker will use them for malicious purposes. In our observation span, we identify 3,128 cases of Private use ASNs involved in sub-MOAS events.

AS relationships

Here we compare AS relationships between ASes involved in the sub-MOAS events. We divide sub-MOAS events in seven buckets according to the ASes involved in them. The seven buckets are Sibling, Customer, InPath, Customer-cone, Provider, Peer and Not found. Sibling means that both ASes belong to a same organization according to the dataset available here [6].

Customer means that sub-ASN is a customer of super-ASN and Provider signifies the inverse of it. Peer specifies peering relationship between ASes. Customer-cone bucket has those events where the sub-ASN can be reached from its super-ASN by traversing customer paths alone. We use inferences done by Luckie et.al in their ASrank dataset [7] to determine provider, customer, peering and customer-cone relations between the AS pairs. InPath bucket is for the events when atleast one of the AS paths towards the prefix originated by sub-ASN has super-ASN in it.

Siblings	Customer	InPath	Customer-Cone	Providers	Peers	Not Found
8.7% 9%	39.7%	9%	1.5%	4.8%	1%	35%

Table 5: Percentage of subMOASes for each bucket

Table 5 shows the percentage of unique sub-ASN–super-ASN pairs involved in all sub-MOAS events observed. We can create a white-list of sub-MOAS events according to the type of bucket they belong to. We can white-list the following buckets:

1: Customer: This case can't be malicious because the super-ASN is letting the announcements pass through it so it is unlikely that super-ASN will let malicious advertisements to propagate beyond its network.
2. InPath: Same argument as the customer bucket can be applied here. The super-ASN will not let announcements to propagate if they are malicious.
3. Customer-cone: Same argument as the customer bucket again. The sub-ASN is indirectly buying transit from its super-ASN .
4. Siblings: Both ASes belong to same organization.

Combining all four above mentioned buckets, we can white-list 58.3% of unique AS pairs observed in sub-MOAS events. If we count every event (including duplicate events), we can white-list 78.44% of total sub-MOAS events seen in two months. As far as the time complexity of this classification is concerned, except inPath, each bucket decision can be taken in O(1). One limitation of this classification is that a hijacker might append such origin ASN before its own ASN, that makes the sub-MOAS event fall in one of the white-listed buckets.
This type of clever ASN prepending can yield false negatives.

Rate of change of sub-MOASes
We analyze the rate by which we observe new sub-MOAS events. Figure 6 shows sub-MOAS events added on time scale. This also counts repeating sub-MOAS events and sub-MOASes spanning multiple prefixes between a same pair of sub-ASN and super-ASN . The arrival of individual sub-MOAS events tends to be bursty in nature and the average number of subMOAS events observed per hour is 107.7. Next we study, how many of these events are new and unique. We define a sub-MOAS event as unique when sub-ASN–super-ASN pair is unique for a set time interval. Since the same type of sub-MOAS events need to be verified once, this type of study helps prefix hijack detection systems in choosing a suitable time window for keeping unique sub-MOAS events in the memory. Within a window, a unique sub-MOAS is event is counted only once even if it occurs repeatedly. Figure 2 shows new sub-MOAS events generated for different window sizes.
Unlike the bursty nature of individual events, the arrival rate of unique events seems to be stable and manageable. The arrival rate for new unique sub-MOAS events for

window sizes of 7 days, 14 days and 21 days was 7, 6 and 5.5 events per hour respectively. Even with a small window size of 7 days, we do not observe too many spikes or new unique events. So by using a small time window of unique sub-MOAS events, we have to consider much less events as compared to keeping track of each event. Another observation is that, as we keep on increasing the window size, the noise around y-axis does not decrease dramatically, which shows that repeating events are grouped close

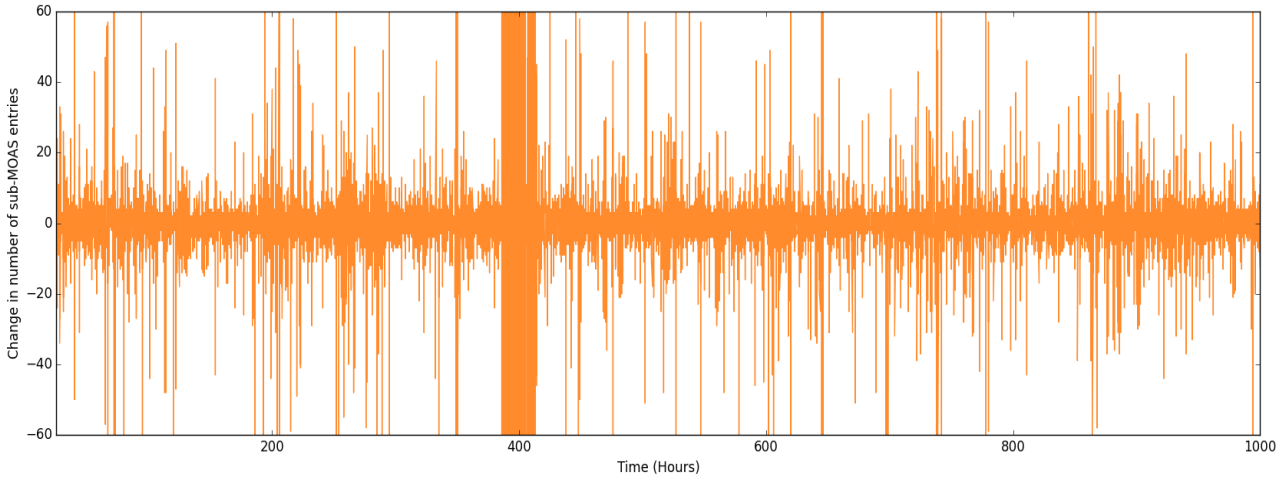

Figure 6: Net change in subMOASes

together on the time axis. Note that some of the spikes remain same across all the window size, we identify these spikes as hijacks based on various properties in section 4.5.

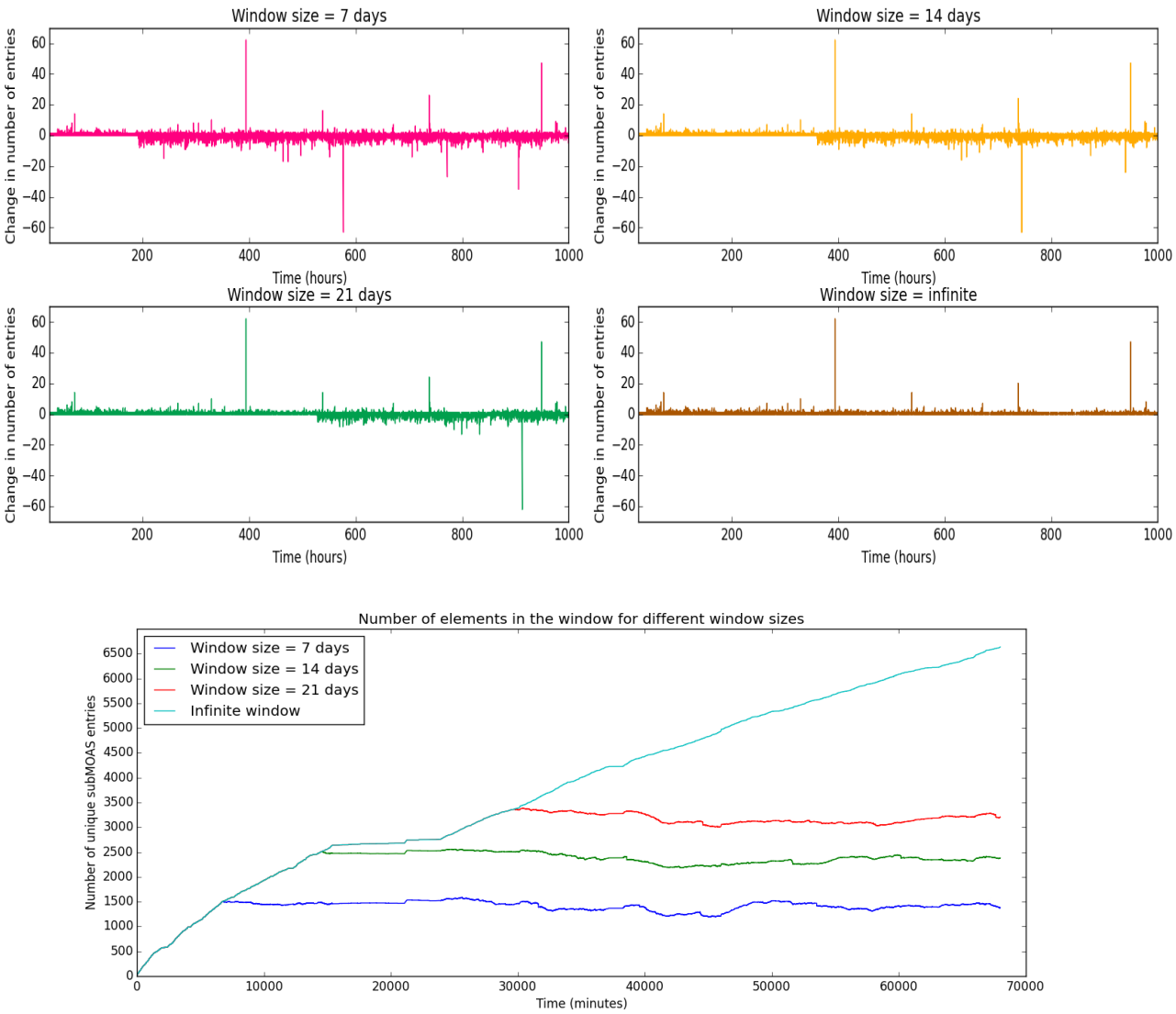

Figure 8: Number of MOASES in a window

Next, we study the number of entries that we have to keep in the memory for different window sizes (Figure 7). Once reaching a threshold, the number of entries remain stable for different window sizes, which shows that arrival rate of new unique sub-MOAS events does not change a lot. Nevertheless, we observing new sub-MOAS events as evident from infinite window curve, so having a small window is necessary for hijack detection systems.

Duration of SubMoas event

Figure 9 shows the distribution of sub-MOAS durations calculated according to the methodology defined. A large number of sub-MOAS events did not finish in our

observation (6,6018). This number is close to the spike of sub-MOASes we observed on day one of our observation period. These sub-MOASes look legitimate as they have been part of the RIBs for a long time (Manual analysis of 20 of those events revealed that 17 of them have been part of RIBs for atleast last two years). For those sub-MOAS events that did finish, most of them finished within an hour (60%). Next we identify some cases of hijack. We tag sub-MOAS events as potential hijacks if they match all of the following criteria:

1. Same ASN being sub-ASN for large number of super-ASNs
2. Sub-MOASes being originated from ISPs that are confined to their own regions. For example, PTCL, a small Internet provider of Pakistan has prefixes only for Pakistan region.
3. We used ASNs whois data to confirm this.
4. Sub-ASNs originating prefixes belonging to other ASNs located in different countries (sometimes different continents). For example, A local ISP from New-Zealand being a sub-ASN for a local ISP from Italy.
5. The sub-MOAS events start close to each other on time axis and have durations similar to each other.

6. The sub-ASN being mentioned as a Spam producing ASN in the blacklists at [8].

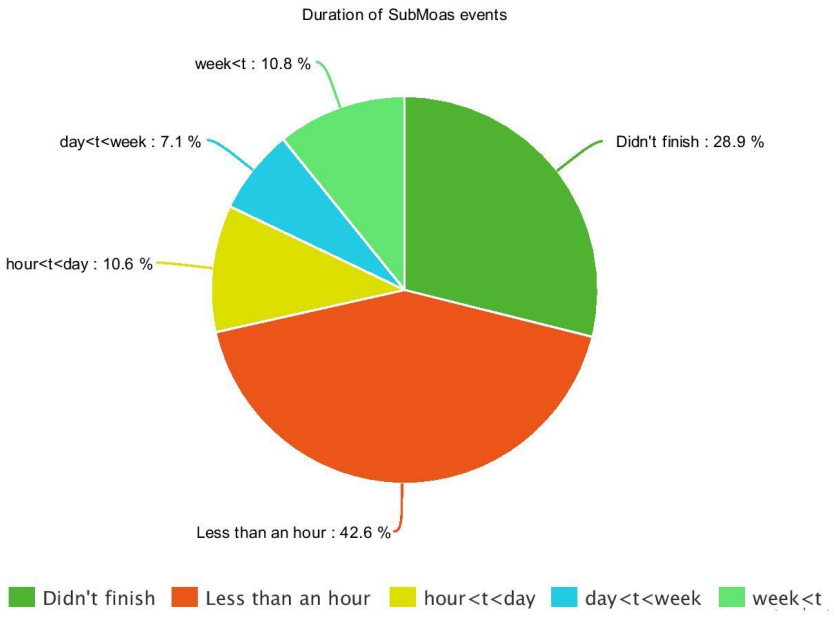

Figure 9: Duration of subMOAS events

Combined, we isolate 1,636 cases of sub-MOAS events as hijacks involving 4 sub-ASNs and 130 super-ASNs. Figure 5 shows the distribution of hijack events in terms of time durations.
The majority of identified hijacks finished in short time durations with 97% finishing within first 12 minutes.

Data Plane Analysis

Even after we have identified suspicious events from control plane, we still need to run traceroutes to further increase our confidence in our decision. We use two sources of vantage points:

1. RIPE Atlas
2. CAIDA Ark

RIPE atlas hosts thousands of probes across the globe

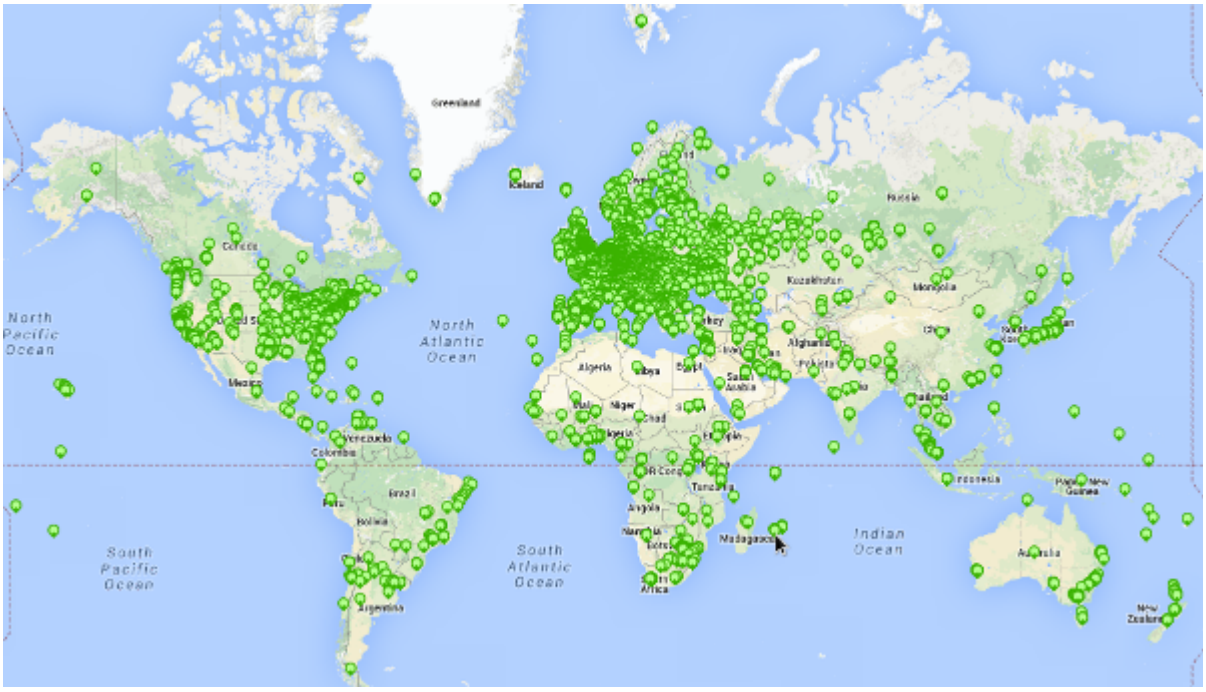

Figure 10: Distribution of RIPE atlas probes

These probes can be used to carry out traceroutes, DNS measurements, ping and other type of network measurements. We used both Ark and RIPE nodes to run traceroutes in response to control plane anomalies

Probe Selection

We picked up probes as close to affected AS as possible. We define AS in subMOAS the subASN. Thus we select probes in neighboring ASes of subASN. We use following scheme to pick up RIPE atlas probes.

1. From affected AS, do a breadth first search on all its neighbours
2. Select at most one ripe atlas probe from the AS
3. Keep on traversing the AS graph till N probes are found

Analyzing results of Traceroutes

We analyzed results of traceroutes which were ran in response to subMOAS control plane anomalies.

We converted IP addresses in the traceroutes to AS path

A subMOAS can have two different type of origin ASes, i.e subASN and superASN.We divided results into following broad categories.

1. **Both Origin ASNs exist**

   Atleast one origin ASN exist in the traceroute path

   a. **Only superASN exist**
   b. **Only subASN exist**
   c. **Both origin ASNs exist**
      i. **Traceroute path sees subASN before the superASN**
      ii. **Traceroute path sees superASN before the superASN**

2. **None of the Origin ASNs exist**

Here we show the fraction of traceroutes which followed each bucket

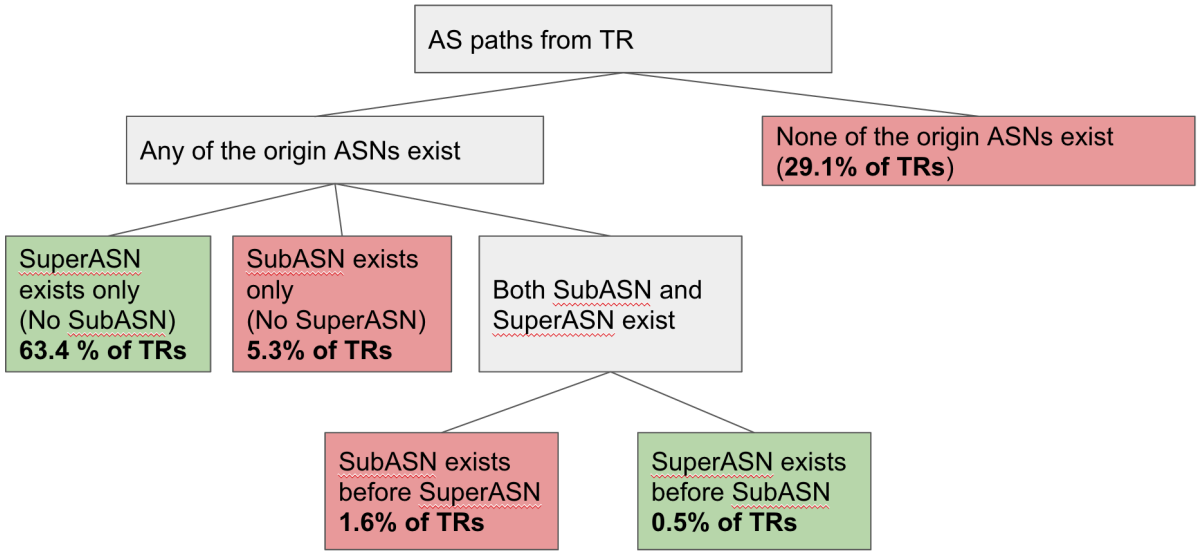

Figure 12: Breakdown of traceroutes

Here we discuss which of these buckets are more likely to be suspicious

Suspicious buckets:

None of the origins exist: This bucket is suspicious as traceroute not reaching its eventual destination is suspicious but this can be due to traceroute information not complete. Reachability of traceroute can never be reliably inferred as the reachability of actual traffic (29.1% of TRs had this property)

SubASN exists only: In SubMOAS, the suspicious ASN in the subASN, so traceroute reaching the subASN only might indicate a hijack (5.3% of traceroutes had this property)

SubASN exists before SuperASN: This can be a case of interception as the traffic reached superASN after passing through subASN, so a rouge subASN can easily sniff the traffic

Here we have described the analysis only for subMOAS. We're currently doing the analysis for other anomalies too, however,The methodology of analysis of other anomalies should be similar too.

Conclusion

We have developed a near real-time BGP hijacks and Interception system. The system uses control plane data from a number of route collectors. We use these router feeds to look for control plane anomalies. Building on insights and datasets from previous works, we isolate suspicious anomalies from the control plane data. Then based on suspicious control plane anomalies, we issue traceroutes to further increase our confidence about an event being a hijack or an interception

References

[1] H. Ballani, P. Francis, and X. Zhang. A study of prefix hijacking and interception in the Internet. In SIGCOMM, 2007.

[2] M. Berman, J. S. Chase, L. Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci, and I. Seskar. GENI: A Federated Testbed for Innovative Network Experiments. Computer Networks, 61:5–23, 2014.

[3] R. Bush, O. Maennel, M. Roughan, and S. Uhlig. Internet optometry: Assessing broken glasses in internet reachability. In ACM IMC, 2009.

[4] X. Cai, J. Heidemann, B. Krishnamurthy, and W. Willinger. Towards an AS-to-organization map. In ACM IMC, 2010.

[5] M. Calder, X. F. Z. Hu, E. K.-B. J. Heidemann, and R. Govindan.
Mapping the Expansion of Google's Serving Infrastructure. In Proceedings of the ACM Internet Measurement Conference (IMC '13), October 2013.

[6] B. Chandrasekaran, M. Bai, M. Schoenfield, A. Berger, N. Caruso, G. Economou, S. Gilliss, B. Maggs, K. Moses, D. Duff, K. Ngâ˘A˘a, E. G. Sirer, R. Weberâ˘A˘a, and B. Wong. Alidade: Ip geolocation without active probing. Department of Computer Science, Duke University, Technical Report, 2015.

[7] K. Chen, D. Choffnes, R. Potharaju, Y. Chen, F. Bustamante, D. Pei, and Y. Zhao. Where the sidewalk ends: Extending the internet AS graph using traceroutes from P2P users. In CoNEXT '09, 2009.

[8] Cisco. BGP Best Path Selection Algorithm: How the Best Path Algorithm Works. Document ID: 13753, May 2012.

[9] L. Colitti. Internet Topology Discovery Using Active Probing. Ph.D.
thesis, University di Roma Tre, 2006.

[10] L. Gao, T. Griffin, and J. Rexford. Inherently safe backup routing with BGP. IEEE INFOCOM, 2001.

[11] L. Gao and J. Rexford. Stable Internet routing without global coordination. Trans. Netw., 2001.

[12] P. Gill, S. Goldberg, and M. Schapira. A survey of interdomain routing policies. ACM CCR, 2014.

[13] P. Gill, M. Schapira, and S. Goldberg. Let the market drive deployment: A strategy for transitioning to BGP security. SIGCOMM'11, 2011.

[14] P. Gill, M. Schapira, and S. Goldberg. Modeling on quicksand: dealing with the scarcity of ground truth in interdomain routing data.
SIGCOMM Comput. Commun. Rev., 42(1):40–46, Jan. 2012.

[15] V. Giotsas, M. Luckie, B. Huffier, and K. Claffy. Inferring Complex AS Relationships. In ACM IMC, November 2014.
[16] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How secure are secure interdomain routing protocols? In SIGCOMM'10, 2010.
[17] T. Griffin, F. B. Shepherd, and G. Wilfong. The stable paths problem and interdomain routing. Trans. Netw., 2002.
[18] G. Huston. Peering and settlements - Part I. The Internet Protocol Journal (Cisco), 2(1), March 1999.
[19] G. Huston. Peering and settlements - Part II. The Internet Protocol Journal (Cisco), 2(2), June 1999.
[20] U. Javed, I. Cunha, D. R. Choffnes, E. Katz-Bassett, T. Anderson, and A. Krishnamurthy. Poiroot: Investigating the root cause of interdomain path changes. In SIGCOMM, 2013.
[21] J. Karlin, S. Forrest, and J. Rexford. Nation-state routing: Censorship, wiretapping, and BGP. CoRR, 2009.
[22] E. Katz-Bassett, C. Scott, D. R. Choffnes, I. Cunha, V. Valancius, N. Feamster, H. V. Madhyastha, T. Anderson, and A. Krishnamurthy.
LIFEGUARD: Practical repair of persistent route failures. In SIGCOMM, 2012.
[23] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In SIGCOMM'10, 2010.
[24] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: A prefix hijack alert system. In Proc. USENIX Security Symposium, 2006.
[25] M. Luckie, B. Huffaker, A. Dhamdhere, and V. Giotsas. AS relationships, customer cones, and validation. In ACM Internet Measurement Conference, 2013.
[26] R. Lychev, S. Goldberg, and M. Schapira. Is the juice worth the squeeze? BGP security in partial deployment. In SIGCOMM'13, 2013.
[27] H. Madhyastha, E. Katz-Bassett, T. Anderson, A. Krishnamurthy, and A. Venkataramani. iPlane Nano: Path prediction for peer-to-peer applications. In Usenix NSDI, 2009.
[28] R. Mazloum, M. Buob, J. Auge, B. Baynat, D. Rossi, and T. Friedman. Violation of Interdomain Routing Assumptions. In Passive and Active Measurement Conference, March 2014.
[29] W. Mühlbauer, A. Feldmann, O. Maennel, M. Roughan, and S. Uhlig. Building an AS-topology model that captures route diversity. In SIGCOMM, 2006.
[30] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang.
Quantifying the completeness of the observed internet AS-level structure. UCLA Computer Science Department - Techical Report TR-080026-2008, Sept 2008.

[31] Quantcast. http://www.quantcast.com.
[32] RIPE ASN Neighbor History. https:
//stat.ripe.net/widget/asn-neighbours-history.
[33] RIPE RIS raw data. http://www.ripe.net/data-tools/ stats/ris/ris-raw-data.
[34] RIPE Network Coordination Center. RIPE Routing Information Service.
http://www.ripe.net/data-tools/stats/ris/routing-information-service.
[35] M. Roughan, W. Willinger, O. Maennel, D. Perouli, and R. Bush. 10 lessons from 10 years of measuring and modeling the Internet's autonomous systems. JSAC, 2011.
[36] Sandvine. Fall 2012 global internet phenomena, 2012.
[37] B. Schlinker, K. Zarifis, I. Cunha, N. Feamster, and E. Katz-Bassett.
PEERING: An AS for Us. In Proc. ACM HotNets, Los Angeles, CA, October 2014.
[38] TeleGeography Submarine Cable Map.
http://www.submarinecablemap.com/.
[39] University of Oregon Route Views Project.
http://www.routeviews.org/.
[40] V. Valancius, N. Feamster, J. Rexford, and A. Nakao. Wide-area route control for distributed services. In USENIX ATC, 2010.
[41] J. Wu, Y. Zhang, Z. M. Mao, and K. Shin. Internet routing resilience to failures: Analysis and implications. In CoNEXT, 2007.
[42]S. Misel, "Wow, AS7007!," 1997. http://www.merit.edu/mail.archives/nanog/1997-04/msg00340.html.
[43]J. Cowie, "Renesys blog: China's 18-minute mystery," 2010.
http://www.renesys.com/blog/2010/11/chinas-18-minute-mystery.shtml.
[44]M. Brown, "Renesys blog: Pakistan Hijacks YouTube," 2008.
http://www.renesys.com/blog/2008/02/pakistan_hijacks_youtube_1.shtml
[45]S. Goldberg, M. Schapira, P. Hummon, and J. Rexford, "How secure are secure interdomain routing protocols," *SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. -, Aug. 2010.
[46]V. Khare, Q. Ju, and B. Zhang, "Concurrent prefix hijacks: Occurrence and impacts," in Proceedings of the 2012 ACM Conference on Internet Measurement Conference, IMC '12, (New York, NY, USA), pp. 29-36, ACM, 2012.
[47]Y. Chi et al., "Cyclops: the as-level connectivity observatory," Sigcomm comput. commun. rev., vol. 38, (5), pp. 5–16, September 2008. DOI: 10.1145/1452335.1452337. Available: http://doi.acm.org/10.1145/1452335.1452337.

[48]M. Lad et al., "Phas: a prefix hijack alert system," in Proceedings of the 15th conference on usenix security symposium - volume 15 (Usenix-ss'06). , Berkeley, ca,

usa: Usenix association, 2006. Available: http://dl.acm.org/citation.cfm?id=1267336.1267347.

[49]Z. Zhang et al., "iSpy: detecting ip prefix hijacking on my own," in Proceedings of the acm sigcomm 2008 conference on data communication (Sigcomm '08). , New york, ny, usa: Acm, 2008, pp. 327–338. DOI: 10.1145/1402958.1402996. Available: http://doi.acm.org/10.1145/1402958.1402996.

[50]C. ZHENG *et al.*, "A Light-weight Distributed Scheme for Detecting Ip Prefix Hijacks in Real-time," in *Proceedings of the 2007 conference on applications, technologies, architectures, and protocols for computer communications* (Sigcomm '07). , New York, NY, USA: Acm, 2007, pp. 277–288. DOI: 10.1145/1282380.1282412. Available:http://doi.acm.org/10.1145/1282380.1282412.

[51]X. Hu and Z. Mao, "Accurate real-time identification of IP prefix hijacking," in Security and privacy, 2007. sp '07. ieee symposium on. May 2007, pp. 3–17. DOI: 10.1109/SP.2007.7.

[52]A Psilov, T Kapela. Stealing The Internet An Internet-Scale Man In The Middle Attack.Defcon 2006, Las Vegas

[53]Pierre-Antoine Vervier, Olivier Thonnard, Marc Dacie. Mind Your Blocks: On the Stealthiness of Malicious BGP Hijacks. NDSS'15.

[54]Anirudh Ramachandran and Nick Feamster. 2006. Understanding the network-level behavior of spammers. SIGCOMM Comput. Commun. Rev. 36, 4 (August 2006), 291-302. DOI=http://dx.doi.org/10.1145/1151659.1159947