Spurious Routes in Public BGP Data

Matthew Luckie
CAIDA / UC San Diego
mjl@caida.org

ABSTRACT

Researchers depend on public BGP data to understand the structure and evolution of the AS topology, as well as the operational security and resiliency of BGP. BGP data is provided voluntarily by network operators who establish BGP sessions with route collectors that record this data. In this paper, we show how trivial it is for a single vantage point (VP) to introduce thousands of spurious routes into the collection by providing examples of five VPs that did so. We explore the impact these misbehaving VPs had on AS relationship inference, showing these misbehaving VPs introduced thousands of AS links that did not exist, and caused relationship inferences for links that did exist to be corrupted. We evaluate methods to automatically identify misbehaving VPs, although we find the result unsatisfying because the limitations of real-world BGP practices and AS relationship inference algorithms produce signatures similar to those created by misbehaving VPs. The most recent misbehaving VP we discovered added thousands of spurious routes for nine consecutive months until 8 November 2012. This misbehaving VP barely impacts (0.1%) our validation of our AS relationship inferences, but this number may be misleading since most of our validation data relies on BGP and RPSL which validates only *existing links*, rather than asserting the non-existence of links. We have only a few assertions of non-existent routes, all received via our public-facing website that allows operators to provide validation data through our interactive feedback mechanism. We only discovered this misbehavior because two independent operators corrected some inferences, and we noticed that the spurious routes all came from the same VP. This event highlights the limitations of even the best available topology data, and provides additional evidence that comprehensive ground truth validation from operators is essential to scientific research on Internet topology.

Categories and Subject Descriptors

C.2.5 [**Local and Wide-Area Networks**]: Internet; C.2.1 [**Network Architecture and Design**]: Network topology

Keywords

AS relationships; routing policies

General Terms

Measurement, Validation

1. INTRODUCTION

Researchers depend on public BGP data to understand the structure and evolution of the Internet's AS topology [9, 17, 19, 6, 10, 16]. Operators of individual ASes voluntarily provide this data by establishing BGP sessions with *route collectors* that record and archive routing data. The two largest collectors are the University of Oregon's Route Views (RV) project [4] and RIPE's Routing Information Service (RIS) [3]. An AS that provides BGP data is providing a *vantage point* (VP) from inside their network as the VP shows routes selected by that AS to reach a set of prefixes. In April 2012, 551 VPs in 342 ASes provided BGP data to RV and RIS.

Beginning 2 February 2012, RV began collecting BGP data from VPs connected to the Telx exchange point in Atlanta, GA (*telxatl*). One of these VPs, 198.32.132.97, provided a view that contained thousands of *spurious routes* until 8 November 2012. AS paths embedded in BGP messages are supposed to represent the *control path* taken from the network that announced (originated) the route; RFC 4271 specifies that an AS_SEQUENCE contains an "ordered set of ASes a route in the UPDATE message has traversed" [20]. We define spurious routes as those not announced by the reported origin AS, or those whose AS path does not report the control path taken by the route. In this work we refer to 198.32.132.97 as a *misbehaving VP* because it reports an unusually large number of spurious routes due to a local condition present inside the AS. In section 4 we establish that the spurious routes injected by this VP were derived from traceroutes conducted by a route optimizer as part of deciding the best paths towards thousands of prefixes.

In this work, we analyze the impact this misbehaving VP had, both on the data collected and on subsequent AS relationship inferences. We also explore methods to automatically identify VPs that supply false views of the AS topology and show that other VPs have also supplied spurious routes. We first discuss background and related work (section 2) and then describe the public BGP data and validation data we use (section 3). Section 4 characterizes the behavior of the misbehaving VP, including details of the thousands of spurious routes reported. It is challenging to assess the impact of spurious routes on the current state of Internet topology research, where the best available data for inference and validation asserts the existence but rarely the non-existence of routes. We can ascertain that this misbehaving VP had only a minor impact (0.1%) on the validation of our most complete set of AS relationship inferences, but this VP caused 1,762 (2.3%) additional p2c inferences in April 2012, most of

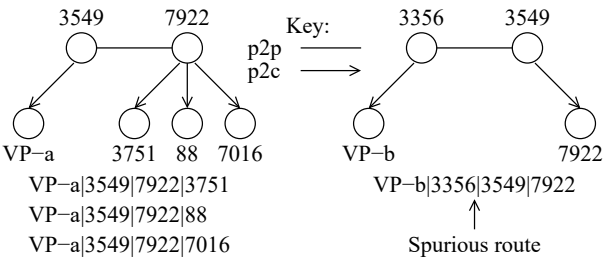

Figure 1: AS paths with a link between 3549 and 7922 from two VPs, observed in public BGP data during October 2012. The link between 3549 and 7922 was p2p, as seen by VP-a. Applying the valley-free assumption to the spurious route from VP-b implies that 7922 was a customer of 3549 because 3549 appeared to provide transit from 7922 to peer 3356. (3356 and 3549 were configured as Tier-1 peers.)

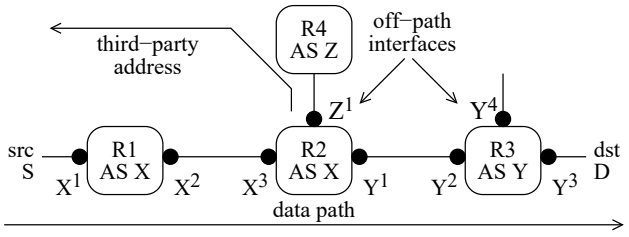

Figure 2: Router-level path between a source S and destination D with off-path interfaces $\mathbf{Z}^1$ and $\mathbf{Y}^4$ labeled. $\mathbf{Z}^1$ is a *third-party address* in this path because it is announced by AS Z which is not in the data path. If R2 replies to a traceroute probe towards D with $\mathbf{Z}^1$, the derived AS path will be false.

which are likely false. Section 5 evaluates possible methods for identifying such anomalies; we find evidence in archived BGP data of four other misbehaving VPs and describe their behavior. Section 6 summarizes lessons we learned.

2. BACKGROUND AND RELATED WORK

AS paths not only represent connectivity between networked organizations, they can be used to infer business relationships between those organizations [9, 7, 2, 16]. Business relationships between ASes, which are typically congruent with their routing relationships, can be broadly classified into three types: customer-to-provider (c2p, or p2c), peer-to-peer (p2p), and sibling-to-sibling (s2s). A p2p relationship provides access to the peer's customers. A c2p relationship gives the customer access to the provider's customers, peers, and providers. Siblings may import and export routes unconventionally between themselves because they are owned by the same organization. Most AS relationship inference algorithms rely on "valley-free" AS paths, an embedded assumption about the rationality of AS routing decisions. Given a p2p or p2c link, the valley-free assumption implies relationships between *downstream ASes* (ASes after a p2p or p2c link) are p2c. The valley-free assumption also implies that there is at most one p2p link in a route, and that most links in a path represent c2p or p2c relationships. A recent survey of operators [11] shows that import and export policies generally follow from the business relationship, but that routing can be more complicated than a naïve AS-level model might imply. Spurious routes disrupt the ability to accurately infer AS relationships because they introduce false links, most of which will be inferred to be p2c, and can cause existing links to have the wrong relationship inferred, i.e. p2p links be incorrectly inferred to be p2c and vice-versa. Figure 1 shows an example from public BGP data: in reality 3549 and 7922 were peers, but the spurious route from VP-b suggested a p2c relationship because 3549 and 3356 were configured as Tier-1 peers, so we incorrectly inferred the 3549-7922 link to be p2c. All BGP-based AS relationship inference algorithms rely on the accuracy of BGP paths to infer relationships between ASes.

Prior work has warned about false links in topology data, particularly when collected using traceroute. False router-level links can be inferred due to per-flow load-balancing, in which more than one path is active from a given source to destination. Classic traceroute, which varies the UDP destination port number or ICMP checksum of successive probe packets, can cause per-flow load-balancers to treat successive packets as distinct flows and forward them along different paths. Consequently, false links can be inferred when successive traceroute probe packets solicit responses from unconnected routers [5, 15]. False links can also be inferred when MPLS is used because the underlying IP path can be obscured between unconnected routers [22, 23, 8].

False AS-level links and paths can also be inferred from traceroute data as an artifact of IP2AS mapping of interfaces collected with traceroute. Figure 2 illustrates just one challenge in using traceroute to infer AS paths: if a router responds to a traceroute probe with an off-path address, and that address maps to a third-party AS (an AS that does not carry traffic on the route), the inferred AS path will be spurious and may contain false links. Zhang *et al.* found the majority of false links in an AS topology data set they derived using traceroute data from four VPs in 2009 were due to third-party addresses [24]. Another contributor to false AS links inferred from traceroute data is an organization using multiple *sibling* ASes to organize their routing; an inferred AS path may differ from the corresponding control path for a route depending on which sibling ASes announce the IP addresses observed in a traceroute path [24]. In addition, the presence of IXP addresses in paths and prefixes originated by multiple ASes [24] inhibit the use of traceroute to infer AS paths.

BGP data should suffer less from false links than traceroute paths because AS paths are formed as part of the process of communicating routes; RFC 4271 specifies that an AS_SEQUENCE contains an "ordered set of ASes a route in the UPDATE message has traversed" [20]. That is, a router prepends its ASN to an existing AS path embedded in a route before announcing the route to a neighbor. However, operators may manipulate AS paths carried in BGP using features implemented by router vendors, and this idealized view of an AS path does not always hold.

For example, there are false links and paths in BGP data due to AS path poisoning [21]. The goal of path poisoning is traffic engineering, specifically to prevent an AS X from selecting a route by inserting X into the AS path announced to neighbors; if X receives a route with its ASN already present in the path, it should discard it to prevent a loop

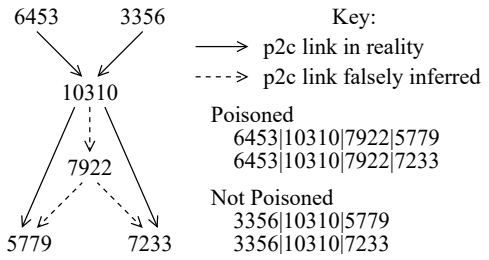

Figure 3: Path poisoning in AS paths observed in public BGP data during April 2012. AS10310 (Yahoo) inserted AS7922 (Comcast) into routes it announced to AS6453 (Tata) to prevent Comcast selecting a path to Yahoo via Tata. This poisoning causes false link and p2c inferences.

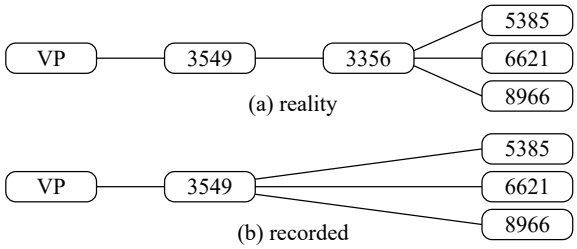

Figure 4: Transition process merging ASes 3549 and 3356 observed in public BGP data December 2012 to May 2013. To transition the VP into 3356's network, 3549 used the BGP "local-as" feature so that the neighbor did not have to reconfigure. Customers of 3356 (a) then appeared to be customers of 3549 (b).

forming [20]. AS path poisoning was used by researchers in LIFEGUARD [13] to automatically localize and repair long-lasting outages. Some poisoning is simple to filter: Y can prevent X from selecting the path by announcing a path with Y on both sides of X, as in "Y X Y" [12]. Because the path contains Y twice, the path can be easily identified as poisoning and filtered out. This style of poisoning was chosen by researchers in the LIFEGUARD work [13] so that it would be clear to other operators that they were poisoning routes. However, an AS is free to insert X anywhere into the path without surrounding the poisoning with its own ASN, and can insert multiple ASes into the path to prevent multiple ASes from selecting the path. For example, figure 3 shows how one AS poisoning a single AS causes multiple false links to be inferred. These cases are difficult to automatically and accurately detect and remain an unsolved problem.

Router vendors provide other features that allow manipulation of AS paths so that an AS path does not reflect the control path. For example, when one ISP purchases another, it might be difficult to transition all customers to one AS in a reasonable time frame due to complicated customer configurations. Figure 4 illustrates an example with the merging of AS3549 (GlobalCrossing) and AS3356 (Level3). Level3's operators used the "local-as" feature when configuring a BGP session with an AS (that also provided a VP) so that the operators of Level3's neighbor (the VP) were not required to modify their BGP configurations. However, the paths recorded by the VP do not reflect the control path taken by the routes. While ASes 3356 and 3549 are operated by the same organization (Level3), the routes are spurious because they do not reflect the control path; in figure 4 neither 5385, 6621, or 8966 had a BGP session configured with 3549. In addition, router vendors allow operators to suppress prepending of an ASN before announcing routes to neighbors. This feature is used by some IXP route server operators to prevent the route-server from prepending the route server's ASN, because the route server does not participate in routing traffic; it merely serves to scalably interconnect networks at the IXP.

In this paper, we introduce a new cause of spurious routes in BGP: autonomous systems that modify AS paths or construct and then inject false routes into public BGP data collection systems. We provide examples of VPs that have done so, including one that did so for nine months in 2012. We evaluate the impact on AS relationship inference, but spurious routes may also impact detection of prefix hijackings [14]. Systems that rely solely on BGP data to detect prefix hijacking events may falsely identify hijacks because spurious routes may be more specifics of a less specific prefix, contain an unexpected AS path, or be originated by an unexpected AS.

3. DATA

3.1 BGP paths

Our process for collating BGP path data is identical to the process we used in our AS relationship inference study [16]; we reuse it here because we evaluate the impact of the spurious routes on AS relationship inferences in section 4. We derive BGP paths from routing table snapshots collected by Route Views [4] and RIPE's RIS project [3]. For each collector, we download one RIB file per day between the 1st and 5th of every month and extract the routes that announce reachability to IPv4 prefixes. We used all paths and not just "stable" paths because backup c2p links are more likely to be included if we use all AS paths, and temporary peering disputes where disconnection occurs may prevent a normally stable path from appearing stable in our five-day window of data. We record all AS paths seen in any of the five snapshots and record the prefixes and BGP origin codes (IGP, EGP, incomplete) seen with each path. We discard paths that contain loops because these are likely due to path poisoning, as well as paths that contain AS-sets because they lack the structure necessary to analyze the AS-level topology. Finally, we compress path padding (i.e. convert an AS path from "A B B C" to "A B C").

3.2 Validation Data

We use the same validation data as collected for [16]. This validation dataset contains 48,276 relationships: 30,770 p2c and 17,506 p2p, collected using multiple sources: BGP communities embedded in route announcements and a dictionary to translate their meaning, RPSL in the RIPE WHOIS database, and direct feedback received through CAIDA's AS-rank website [1]. We use this data to estimate the impact of this VP on the inference of AS relationships. Our validation data primarily contains inferences for links that exist in the AS topology because they are mostly derived from BGP communities and RPSL, which detail the routing policy for links that exist. The third source, CAIDA's AS-rank website [1], allows operators to assert the non-existence

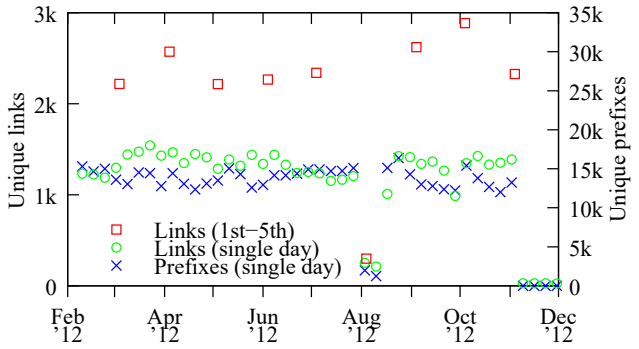

Figure 5: Number of unique links and prefixes provided by the misbehaving VP to the RV *telxatl* collector over time, both for single days (every Wednesday from February to December 2012) and accumulated from snapshots from the 1st to 5th of each month. Most single snapshots have >10k spurious prefixes (>99% of them of /19 length) and >1k spurious links. Accumulating links across five snapshots results in 2k-3k spurious links because the optimizer injects routes depending on current load. The numbers of unique prefixes and unique links are correlated. We are unsure why the VP provided relatively few spurious routes in early August. As with other time periods, the vast majority of the spurious routes it created were for /19 prefixes.

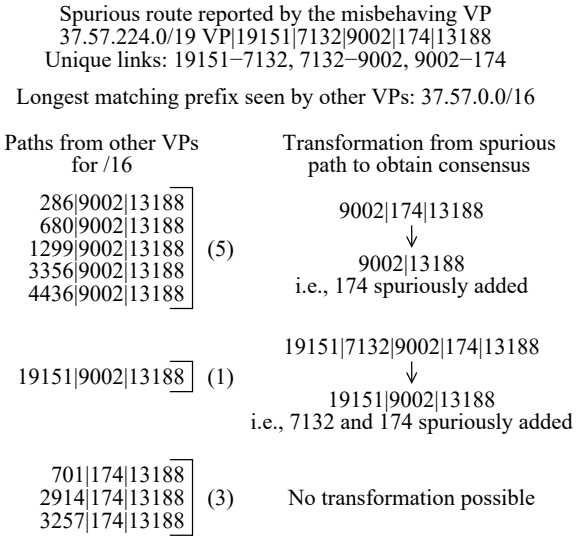

Figure 6: Exploring spurious links by classifying differences between paths advertised by the misbehaving VP and those advertised by other VPs for the longest matching prefix. The view provided by most VPs is consistent: 174 does not appear between 9002 and 13188. Because 19151 provides a view we also infer that 7132 does not appear between it and 9002.

of links, but such assertions are rare: we had 8 non-existence assertions in the 6 months where we posted AS relationships data corrupted by the misbehaving VP, and only two of these assertions were caused by the misbehaving VP.

4. CHARACTERIZATION OF 198.32.132.97

In this section, we characterize the spurious paths advertised to *telxatl* by the misbehaving VP and quantify their impact on the graph. We first became aware of the problems caused by this VP through the interactive feedback functionality of our public repository of AS relationships. When we received feedback that we incorrectly inferred a provider of an AS, we contacted the submitter of this feedback to query the validity of the specific paths we used to derive the provider inference. Two independent operators identified false paths that turned out to have been reported by the same VP, including: (1) links that did not exist, and (2) some links in a path that did not exist. On further investigation, we found this VP to be providing a significant number of unique links and prefixes, most of which we then suspected were false. The misbehaving VP was a stub AS between June and September 2012, and had a single downstream AS for the remainder of 2012.

Figure 5 shows typically more than 10k prefixes and 1k links were unique to this VP between February and November 2012, and the number of links and prefixes unique to this VP were correlated. In addition, there was limited overlap in links when accumulating five snapshots, as we do when inferring AS relationships [16], as there were at least 2k unique links for the 1st to 5th between March and November. We contacted the operator of this VP and learned the cause of these spurious paths was a commercial route optimizer they deployed, which leaked routes into the view provided to *telxatl*. In order to infer the best route in terms of loss, jitter, and hop count, the route optimizer used traceroute to probe available paths in iBGP through different neighbors to a given destination. The AS paths injected into BGP were (falsely) inferred from these traceroutes. The route optimizer created sets of more specific prefixes from a set of less specific prefixes received from different neighbors. The route optimizer then tried to balance traffic load among available upstreams rather than use a single best path. The vast majority (>99%) of prefixes from the misbehaving VP not seen by any other VP were of /19 granularity and were more specifics of another prefix. Because an AS originating a prefix may also announce more specifics of a less specific prefix, our estimation in figure 5 of the number of prefixes the misbehaving VP created is a lower bound.

All the spurious routes had an incomplete origin code in BGP. The origin code reports how the route was learned [20]. An incomplete origin code, signified with a '?', indicates the route was learned from means other than from BGP operating at the origin AS, such as the route being redistributed from another protocol into BGP. Most routes are learned from the origin announcing the route with BGP to its neighbors, signified with a 'i'. The VP stopped exporting routes from the optimizer to the collector on 8 November 2012, after Route Views operators contacted the VP's operator.

Figure 5 shows that the AS paths reported by the misbehaving VP contained many thousands of links that were not seen by other VPs. However, there are serious visibility limitations when using BGP data, particularly the lack of visibility of p2p links [18]. The links seen only by the misbehaving VP may exist; however, we gather evidence many of these unique links are spurious and confirm our findings by obtaining ground truth from operators. Figure 6 pro-

	Excluding	Including
P2P PPV / TPR	98.7% / 99.3%	98.7% / 99.2%
P2P inferences	48003	48702
P2C PPV / TPR	99.6% / 99.3%	99.5% / 99.3%
P2C inferences	78079	79841

Table 1: Impact of including paths from the misbehaving VP on AS relationship inference for April 2012. We show the Positive Predictive Value (PPV) and True Positive Rate (TPR) of our inferences. This misbehaving VP barely impacts (0.1%) validation of our AS relationship inferences.

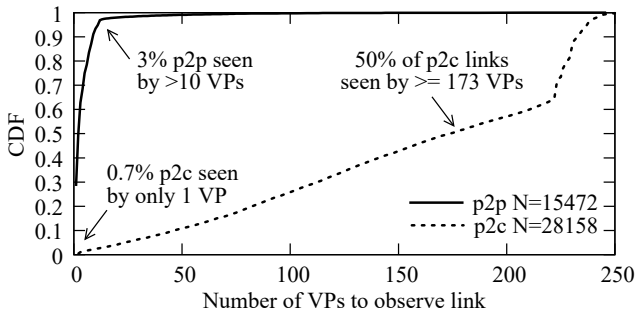

Figure 7: For links in our AS relationship validation dataset (section 3.2), the number of VPs that report that link (by type) in BGP data for April 2012. Only 0.7% of p2c links were seen by one VP.

vides a representative spurious route from the misbehaving VP. The prefix attached to the route is a /19; however, no other VP observes the prefix – the longest matching prefix is a /16 which is observed by nine other VPs. In addition, the 19151–7132, 7132–9002, and 9002–174 links are reported only by the misbehaving VP. We compared the paths in these routes with paths reported by other VPs for the corresponding longest matching prefix, using a method similar to Zhang *et al.* [24]; we infer transformations from the traceroute-inferred to the BGP-observed AS path. Comparing the paths reported for the /16 with the path reported for the /19, we notice the other VPs reported a path that includes some common ASes, but that the misbehaving VP also includes other ASes in a path formation not seen by the first six other VPs in figure 6. We interpret the first five paths as suggesting AS174 was inserted into the path reported by the misbehaving VP, and the sixth path as suggesting that AS7132 was also inserted by this VP. These six paths provide context for where the three unique links were formed by the VP. These links were likely falsely inferred due to the misbehaving VP observing third-party addresses [24] in the traceroutes executed by their route optimizer.

Table 1 shows the impact of including the misbehaving VP on our AS relationship inferences by evaluating the validation of our AS relationship inferences on two sets of paths: one that excludes the VP, and the other that includes the VP. This VP barely impacts (0.1%) our validation of our AS relationship inferences using CAIDA's algorithm for April 2012 [16]; the TPR of our p2p inferences and the PPV of our p2c inferences both reduced by 0.1% when including the VP. Yet, the VP induced 1,762 (2.3%) more p2c inferences and 699 (1.5%) more p2p inferences, most of which

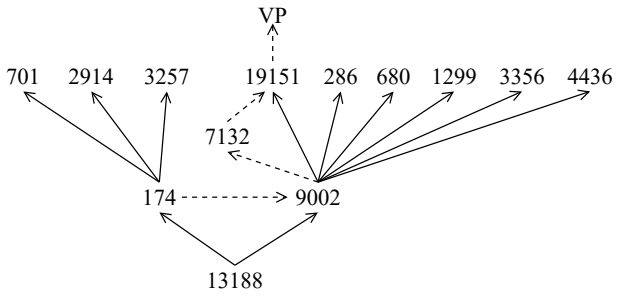

Figure 8: Triangles inferred using paths in figure 6. The triangles are formed using a path from the misbehaving VP to the same prefix. For the triangle to exist in reality, an AS must receive the same prefix with different paths at different points in their network and then advertise each to different neighbors.

are spurious either because the links do not exist, or the relationship inference for them is incorrect. Figure 7 shows the visibility of the p2c and p2p links contained in our validation data; only 0.7% of p2c links were seen by one of the 551 VPs, suggesting it unlikely a single VP would observe 2.3% more p2c links. The disparity between the minor impact on our validation and the large number of additional link inferences is because these links are not covered well in our validation data (section 3.2). Because they do not exist in WHOIS or BGP communities sources, the only reason we know of these spurious links was through direct feedback obtained through our public-facing website. In addition, the feedback we obtained from our website was biased towards reporting incorrectly inferred provider relationships [16].

We were able to confirm some spurious links and relationship inferences through interactions with four network operators. We contacted additional operators, but they were unable to help because of the time difference between when this VP was providing spurious data and when we contacted them in March 2013. A small transit provider supplied ground truth for five spurious links; all five were peers, including two we inferred to be customers. A large transit provider (LTP) provided ground truth for 14 spurious links, none of which existed in reality. Our AS relationship inference algorithm inferred 12 to be customers and two to be peers. Nine of the 12 spurious customers were indirect customers (customers of customers) of the LTP; four of these were directly connected to the LTP in spurious paths through the removal of sibling ASes by the misbehaving VP (because it derived AS paths from traceroute, see section 2). Another small transit provider explained that two providers we inferred were false; neither was connected to their AS. Finally, a large access network provider explained that (1) two inferred providers were actually peers, and (2) many inferred customers were actually indirect customers, falsely inferred as direct customers because their sibling ASes in the control paths did not appear in the spurious paths.

5. SANITIZING BGP DATA

A common pattern emerging from section 4 is that false links cause triangles to be formed in a graph representing advertised paths towards the same prefix. Figure 8 shows the two triangles formed for the prefix in figure 6. However,

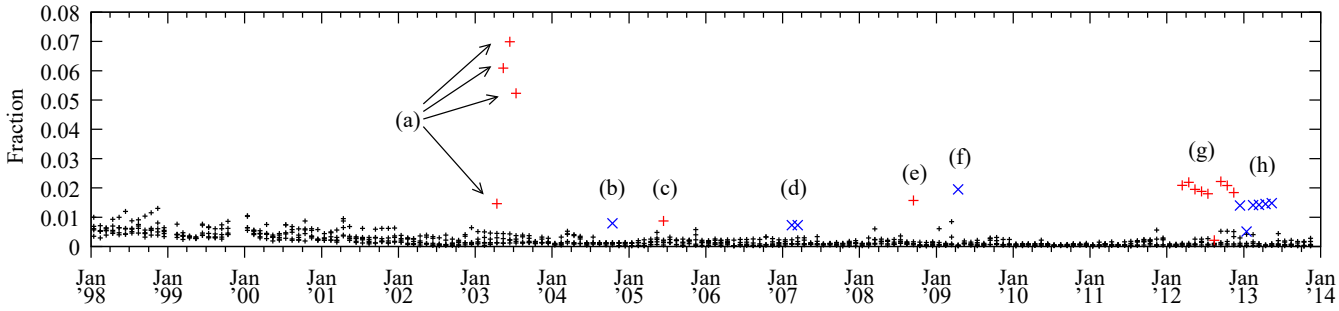

Figure 9: Fraction of p2c links unique to a VP for each month, for the five VPs contributing the most unique p2c-inferred links. Only 0.7% of p2c links in our validation data were seen by one VP (figure 7). The VPs that contribute the most unique links that are inferred to be p2c typically contribute fewer than 0.3%, though there are outliers, particularly due to misbehaving VPs. Prior to 2002, there were few deployed BGP collectors and connected VPs, so many p2c links were observed by just one VP. Table 2 summarizes the labeled points; these points have different symbols to distinguish them from points that are not outliers.

	Peer IP	Collector	Date
a	167.142.3.6	Routeviews	May '03 – Jun '03
	Removal of ASes from paths		
b	168.209.255.2	RIPE rrc00	Oct '04
	Mis-inferring p2p links as p2c		
c	193.203.0.7	RIPE rrc05	May '05
	Mis-inferring p2p links as p2c		
d	198.32.160.103	RIPE rrc11	Feb '07 – Mar '07
	AS701 removing sibling ASes 702, 703, 14551		
e	207.246.129.4	Routeviews3	Sep '08
	AS paths derived from traceroute		
f	200.219.130.10	RIPE rrc 15	Apr '09
	Mis-inferring p2p links as p2c		
g	198.32.132.97	RV Telxatl	Mar '12 – Nov '12
	AS paths derived from traceroute		
h	198.32.176.13	RV ISC	Dec '12 – May '13
	AS3549 removing sibling AS 3356		

Table 2: Outliers labeled in figure 9, and their causes. Of the 8 labeled outliers, three are due to the VP injecting false paths, and two are due to organizations removing their sibling ASes before the routes are announced to a peer. The other three are likely due our AS relationship inference algorithm incorrectly inferring p2p links as p2c.

triangles are widespread in the AS topology, limiting its utility as a filtering mechanism. In April 2012, 401,075 prefixes covering 99.3% of the announced IPv4 address space contained at least one triangle using the paths provided by VPs at RV and RIS. Most triangles involve large transit providers selecting a different path at different points in their network toward the same prefix, likely due to hot potato routing or traffic engineering policies of the origin. Because many triangles caused by the misbehaving VP involved the insertion or deletion of an AS immediately upstream of the origin, we computed the subset of paths with triangles between an origin and its immediate neighbors, which reduced the number of prefixes with triangles to 42,206, covering 28.5% of the address space. We conclude that the existence of triangles in a graph representing advertised paths towards the same prefix is an insufficient signal to identify misbehaving VPs.

A second common pattern emerging from section 4 is that the spurious paths contain links that are reported only by one VP, and most (71.6%) of these links are inferred to be p2c. Indeed, figure 7 shows that only a tiny fraction (0.7%) of p2c links in our validation data were seen by only one vantage point. Figure 9 shows the fraction of all p2c links that were unique to a given VP for each month beginning in January 1998, and table 2 lists our inference of the cause behind the labeled points. We manually investigated the labeled points; of the eight labeled points, three indisputably are spurious: (1) point g, which corresponds to the misbehaving VP described in section 4, (2) point a, the result of a VP that retained the neighbor and origin ASes in the path, but no ASes in between, and (3) point e, which appears also due to a route optimizer that announced paths derived from traceroute. For points d and h we inferred the upstream to be stripping a sibling AS from the path before announcing it downstream; point d corresponds to AS701 deleting ASes 702, 703, and 14551 from paths before announcing them to a peer, and point h corresponds to (previously GlobalCrossing's, now Level3's) AS3549 removing Level3's AS3356 from paths before announcing them to a peer. Figure 4 illustrates the behavior corresponding to point h. The other three labeled points in figure 9 (b,c,f) are outliers, and manual inspection suggests they are p2p links that our AS relationship inference algorithm [16] incorrectly inferred as p2c.

Figure 9 suggests that we could derive a method to automatically filter spurious VPs, where VPs contributing a large fraction of unique inferred p2c links are filtered. We are hesitant to define a threshold in this paper because the misbehaving VPs had a wide variation in the fraction of unique inferred p2c links. Including only stable paths (those seen in all five snapshots) would have avoided most spurious routes for the two route optimizer cases (e and g) because the optimizer selects paths based on their measured performance and current demand.

6. LESSONS LEARNED

Researchers trust that the paths received in BGP accurately convey the AS path taken by the route to reach the VP, yet there is little critical assessment of the accuracy of BGP AS paths compared with the scrutiny placed on AS paths derived using traceroute data (e.g. [23, 24]). Our

study is a first attempt to analyze the impact of vantage points injecting spurious routes into public BGP data collection systems. Our methodology focused solely on the prevalence of inferred p2c links observed by only one VP; yet, misbehaving BGP speakers could be present in the core of the topology and their spurious routes spread much more widely. We therefore believe our results are a conservative lower-bound on the prevalence of misbehaving actors in public BGP data.

All sources of macroscopic Internet topology measurement data have epistemological limitations. In the case of BGP, a single vantage point can inject many false AS links into a data collection. We showed that a misbehaving VP introduces mostly p2c links not observed by other VPs, though we stopped short of recommending a method to detect such false links, because illegitimate behavior from misbehaving VPs is difficult to automatically distinguish from limitations of our AS relationship inferences. The impact of such false routes is also challenging to determine, since our validation data primarily includes assertions of existing rather than non-existing relationships. The misbehaving VP had only a minor impact on the validation of our AS relationship inference algorithm, but this VP caused 1,762 additional p2c inferences, most of which are likely false. We only discovered this misbehavior because two independent operators corrected inferences, and we noticed some spurious inferences were made from routes reported by the same VP. This event highlights the limitations of even the best available Internet topology data, and provides additional evidence that comprehensive public-facing ground truth validation, including of non-existent routes, is essential to detecting certain types of false inferences. We believe the community needs to place a much higher value on such public-facing validation.

Acknowledgments

We thank the anonymous reviewers for their feedback. The work was supported by U.S. NSF grant CNS-0958547, DHS S&T Cyber Security Division (DHS S&T/CSD) BAA 11-02 and SPAWAR Systems Center Pacific via N66001-12-C-0130, and by Defence Research and Development Canada (DRDC) pursuant to an Agreement between the U.S. and Canadian governments for Cooperation in Science and Technology for Critical Infrastructure Protection and Border Security. This material represents the position of the author and not of NSF, DHS, or DRDC.

7. REFERENCES

[1] CAIDA's AS-rank project. `http://as-rank.caida.org/`.

[2] Internet Topology Collection. `http://irl.cs.ucla.edu/topology/`.

[3] RIPE (RIS). `http://www.ripe.net/ris/`.

[4] University of Oregon Route Views Project. `http://www.routeviews.org/`.

[5] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira. Avoiding traceroute anomalies with Paris traceroute. In *IMC*, Oct. 2006.

[6] A. Dhamdhere and C. Dovrolis. Twelve years in the evolution of the Internet ecosystem. *IEEE/ACM Transactions on Networking*, 19(5), 2011.

[7] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, and kc claffy. AS relationships: Inference and validation. *CCR*, 37(1):29–40, Jan. 2007.

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

[9] L. Gao. On inferring autonomous system relationships in the Internet. *IEEE/ACM Transactions on Networking*, 2001.

[10] P. Gill, M. Schapira, and S. Goldberg. Let the market drive deployment: A strategy for transitioning to BGP security. In *SIGCOMM*, Aug. 2011.

[11] P. Gill, M. Schapira, and S. Goldberg. A survey of interdomain routing policies. *CCR*, 44(1):29–34, Jan. 2014.

[12] G. Huston. Exploring autonomous system numbers. *The Internet Protocol Journal*, 9(1):2–23, Mar. 2006.

[13] E. Katz-Bassett, C. Scott, D. R. Choffnes, Ítalo Cunha, V. Valancius, N. Feamster, H. V. Madhyastha, T. Anderson, and A. Krishnamurthy. LIFEGUARD: Practical repair of persistent route failures. In *SIGCOMM*, 2012.

[14] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: A prefix hijack alert system. In *USENIX Security*, Aug. 2006.

[15] M. Luckie, A. Dhamdhere, k.c. claffy, and D. Murrell. Measured impact of crooked traceroute. *CCR*, 41(1):14–21, Jan. 2011.

[16] M. Luckie, B. Huffaker, A. Dhamdhere, V. Giotsas, and k claffy. AS relationships, customer cones, and validation. In *IMC*, Oct. 2013.

[17] W. Mühlbauer, S. Uhlig, B. Fu, M. Meulle, and O. Maennel. In search for an appropriate granularity to model routing policies. In *SIGCOMM*, Aug. 2007.

[18] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang. In search of the elusive ground truth: The Internet's AS-level connectivity structure. In *SIGMETRICS*, 2008.

[19] R. Oliveira, B. Zhang, and L. Zhang. Observing the Evolution of Internet AS Topology. In *SIGCOMM*, Kyoto, Japan, Aug. 2007.

[20] Y. Rekhter, T. Li, and S. Hares. A border gateway protocol 4 (BGP-4). RFC 4271.

[21] 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.

[22] R. Sherwood, A. Bender, and N. Spring. DisCarte: a disjunctive Internet cartographer. In *SIGCOMM*, Seattle, WA, USA, Aug. 2008.

[23] W. Willinger, D. Alderson, and J. C. Doyle. Mathematics and the Internet: a source of enormous confusion and great potential. *Notices of the American Mathematical Society*, 56(5), May 2009.

[24] Y. Zhang, R. Oliveira, H. Zhang, and L. Zhang. Quantifying the pitfalls of traceroute in AS connectivity inference. In *PAM*, 2010.