Internet optometry: assessing the broken glasses in Internet reachability

Randy Bush
IIJ
randy@psg.com

Olaf Maennel
T-Labs
olaf@maennel.net

Matthew Roughan
University of Adelaide
matthew.roughan@adelaide.edu.au

Steve Uhlig
T-Labs
steve@net.t-labs.tu-berlin.de

ABSTRACT

Reachability is thought of as the most basic service provided by today's Internet. Unfortunately, this does not imply that the community has a deep understanding of it. Researchers and operators rely on two views of reachability to assess it: control/routing- and data-plane measurements.

Both control- and data-plane measurements suffer from different biases and have their own limited visibility. In this paper, we illustrate some of these biases, and show how to design controlled experiments which allow us to "see" through the limitations of previous measurement techniques. For example, we discover the extent of default routing and its impact on reachability. This explains some of the unexpected results from studies that measured the correlation between the control and the data plane.

However, not all limitations of visibility given by routing and probing tools can be compensated for by methodological improvements. As we show in this paper, some of the limitations can be carefully addressed when designing an experiment, e.g. not seeing the reverse path taken by a probe can be partly compensated for by our methodology, called *dual probing*. Compensating for other biases through more measurements may not always be possible. Therefore, calibration of expectations and checks of assumptions are critical when conducting measurements that aim at making conclusions about topological properties of the Internet.

1. INTRODUCTION

It may be too obvious to mention, but the fundamental service of the Internet is any-to-any connectivity. If I connect to the Internet at any point, I should be able to reach any other host, though that host may, of course, reject my advances. Much recent Internet research concerns advanced features of the Internet, quality of service, mobility, ... However, we show in this paper that there is still a great deal to learn about the basic *reachability* service of the Internet.

We believe that our poor knowledge of reachability is mainly due to: 1) limitations of the data often used to assess reachability, and 2) poor understanding of the impact of those limitations on claims about data-plane conditions. First, BGP (Border Gateway Protocol) observations only see paths towards the originating ASes (Autonomous Systems), they have no wider vision. Increasing the number of BGP vantage points adds less visibility than one might wish [1]. Second, obtaining adequate coverage with active probes requires being able to reach and get responses from all of the Internet, especially the edge. Third, current tools, e.g., traceroute, only yield information about the forward path from the probing site toward the destination. Obtaining reverse paths using the record route option and by correlating traceroutes, as in [2], is not a general solution to the problem.

This paper reports a series of experiments that illustrate the limitations and biases that arise when trying to assess data-plane reachability from control-plane observations. Our first experiment shows that popular BGP observation points do not see enough to assess the reachability of a prefix. Even very simple data-plane measurements give a much better view than current BGP observations. Our second experiment shows that default routing is used widely in the Internet, particular at stub, but also in transit ASes. The unexpected prevalence of default routing makes reachability as seen from the data plane quite different from prefix visibility in the control plane. The resulting property of "reachability without visibility" is fundamental, and would occur even if we had BGP monitors in every AS! It has far-reaching implications, e.g., we believe it explains the seemingly anomalous results from [3], namely the unexpected differences between the data- and control-plane measurements. Our third experiment illustrates the power of a probing technique, *dual probing*, that leverages the comparison of probing initiated from different parts of the address space. Dual probing first sends probes from well established, *anchor*, address space, and compares it to the results of probes from a *test* address space. Using probes from an anchor address space reduces the chances of misinterpretation of the measurements made using the test address space. Even a powerful probing methodology, such as dual probing, suffers from measurements artifacts and limitations that need to be addressed.

Throughout this paper, we shed light on why the relationship between the data and control plane are so often counter-intuitive [3], and also why researchers and operators should pay more attention to what their measurements really say. The main conclusion of our work is that the visibility available both from the control plane and from popular active measurement techniques is insufficient to make strong claims about the data plane. We insist on the fundamental nature of this issue, which questions much previous literature which made claims about the data plane only from observations of the control plane. This does not mean that control-plane and data-plane behaviors are not related, but rather that control-plane observations need to be very carefully understood before drawing conclusions about data-plane conditions.

The other theme of this paper is the construction of careful *controlled* experiments. The nature of these experiments may seem diverse, because the construction of each experiment is targeted at providing insights into particular properties of the network. We do not "go fishing". We construct experiments where we can answer particular questions, sometimes as a result of a previous experimental result. Our goal is to remove the ambiguities from our hypothesis. For instance, our initial experiment suggested that reachability extended beyond visibility, and so our second experiment was targeted at understanding the potential causes of this property. We argue that this approach should be more widely adopted in Internet measurement research. However, measurements always have problems, and for this reason it is important to continue to test and calibrate.

The paper is structured as follows. First, we give an example of the limitations of BGP observations: we advertise a $/25$-prefix, which we expected to propagate very poorly and then compare BGP observations, with active reachability tests (Section 2). The unexpected reachability of the $/25$ led us to measure the fraction of ASes that use default vs. the faction of ASes that use default-free routing, which we report in Section 3. We then use the AS path poisoning technique to discover hidden upstream providers in Section 4. We present in Section 5 a new probing methodology, called *dual probing*, that takes advantage of active probing initiated from different parts of the address space. In the final part of the paper, we discuss three specific issues of active probing tools, namely the topological coverage of measurements (Section 6.1), the IP to AS number mapping (Section 6.2) and the types of probes to be used (Section 6.3).

2. HOW FAR DOES A $/25$ PROPAGATE?

Anecdotally, most providers filter prefixes more specific than $/24$ to bound the number of routes in the global Internet and reduce grazing of the commons with announcements of overly specific prefixes to control or hijack traffic.

As the starting point for this paper, we sought to test whether such filters are as prevalent as commonly presupposed. We advertised a $/25$ prefix from AS 3130 on June 22nd 2008, and measured its reachability across the Internet via both control-plane and data-plane measurements. The results were so inconsistent that it highlighted the key problem of this paper that control-plane measurements are insufficient to measure data-plane reachability.

The standard means of predicting reachability used by both operators and researchers is to look at the control plane via public BGP observation points and private looking glasses to see where the BGP announcement has propagated and what paths are available. We did the same. RouteViews, and RIPE/RIS route monitors saw the prefix in 11 locations out of 615 RIPE BGP feeds. This matched our expectation that a $/25$ would be severely filtered and would not propagate far.

Our data-plane measurements were based on pinging a large set of IP addresses spread widely across the Internet (see Section 6.3). We used an IP address from the $/25$ as the source of the ping packets. Receiving a ping-response therefore indicates that the ping target can reach our prefix (obviously, no covering prefix was announced). No response could mean that the IP might be down, or the pinged IP might not have a path towards the $/25$-address space, and so we only draw conclusions from the positive responses.

To our surprise, we found $1,024$ ASes that had usable connectivity back to our /25. This represented nearly 5% of all the ASes visible in this experiment.

Even more interesting, all of the BGP observation points which observed announcements of the $/25$ prefix were within 3 AS-hops of the origin. Figure 1 shows for the distribution (the curve indicated by "BGP"). This matched our intuition that most BGP observation points are in the "center" of the Internet (near our announcement). The prefix was announced by AS 3130 which has two tier-1 upstream providers. Tier-1's, and other large transit providers, often have their own BGP observation points. The $/25$ was not propagated to many observation points, and those which saw the specific prefix were in the center. However, our data-plane measurements showed that a significant proportion of the Internet could reach the address space of the prefix, particularly the customers of that tier-1 that accepted the prefix and their customers.

Moreover, we used traceroute toward the pingable target IP addresses to measure the number of AS hops from the origin of the $/25$. The results in Figure 1 show that the number of hops to the $/25$ was not much different from the classic number of hops to the broader set of ASes observed in the study. There were a significant number of ASes 5 hops away that could still reach our $/25$!

In every respect, the results show a clear difference between control- and data-plane measurements of reachability. Control-plane measurements provide predictions of reachability, whereas data-plane measurements show real reachability, so the latter need to take precedence.

There are two likely reasons for the difference: (i) the prefix propagated further than expected on the control plane to sites which were not visible from the standard BGP monitors, and (ii) default routes provided effective connectivity to some ASes despite the fact that they never learned of our prefix. Of particular interest, over 75% of those ASes with data-plane reachability were stub ASes, and as default routing is naively presumed to be more common in stubs, we sought to investigate this cause further.

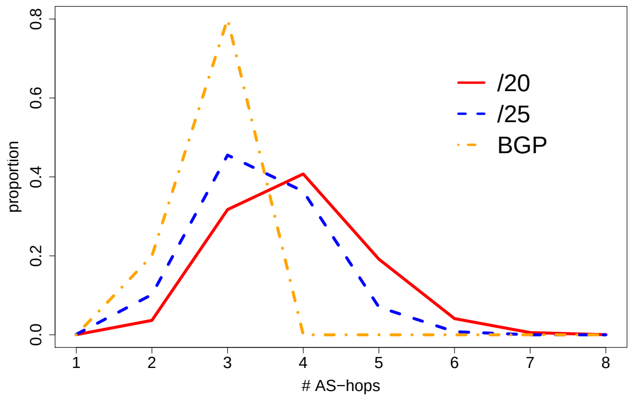

Figure 1: Distribution of the number of AS-hops to the $/25$ reachable ASes, to $/20$ and from BGP monitors.

3. UTILIZATION OF DEFAULT ROUTING IN THE INTERNET

The previous experiment suggested that default routes may cause part of the discrepancy between control- and data-plane measurements. So in this section, we shall examine the prevalence of default routing in the Internet.

We use AS-path poisoning [4] to measure the extent to which default routing is used. Figure 2 illustrates the process. Our test box in AS 3130 announced a set of experimental prefixes p to its upstream tier-1 provider. We announced these prefixes p with poisoned AS paths, i.e. paths containing the AS number of an AS we want to receive the test prefix and drop it. In that way, we can be sure that this AS does not have this prefix in its BGP routing tables. For example, if AS 2 is to be poisoned, we announce ”3130 2 3130”. When AS 2 receives prefix p, it sees its own AS number in the AS path, and drops the announcement because of BGP loop prevention. So, unless AS 2 has a default route, it should not be able to reach an IP address inside prefix p after receiving the poisoned route.

AS path poisoning tests were conducted from from Saturday, April 18 2009 until Friday, May 1 2009. During that 13 day period, we tested $25,780$ ASes for their use of defaults. We use the address space $98.128.0.0/16$[1] sliced into $/24$'s and announced as many sub-prefixes in parallel to be able to cover the large proportion of the Internet that we do. We could not probe all known ASes due to the difficulties finding pingable IP addresses (see Section 6.3), and because some operators explicitly asked us not to probe their AS (which we respect in all cases).

Each experiment was conducted as follows: we use a look-ahead test (from unpoisoned address space) to check that the target AS was reachable before poisoning. We then withdraw the test-prefix, and wait for 1.5 hours to reduce the possible influence of route flap damping. We then announce the (poisoned) prefix, and wait for 20 minutes to allow it to propagate. We then start testing the poisoned AS using pings from the test-address space to our set of target IP addresses in the relevant AS. The time necessary to probe varies, as we have to wait for the ping timeouts if probing a default-free AS and to probe some IP addresses several times to compensate for packet loss. A typical test run took between 2 and 3 hours, but remember that we could conduct a series of such tests in parallel to allow for wide coverage.

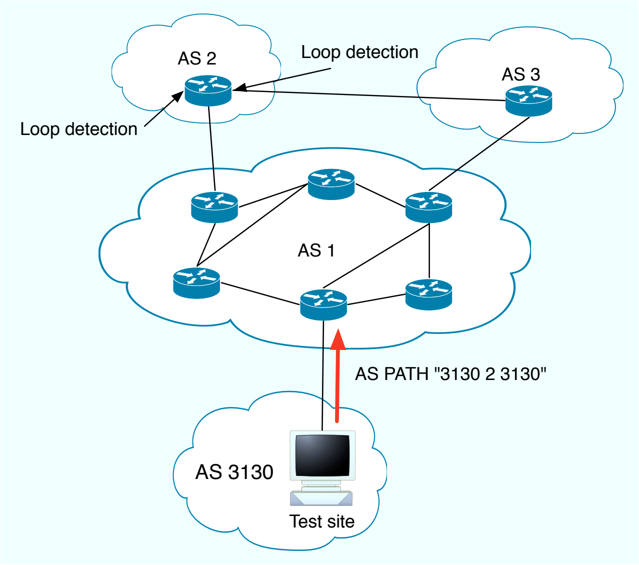

Figure 2: Testing an AS for default routing with AS path poisoning.

We also compared test probes of the target IP addresses from unpoisoned address space to check that these addresses were consistently available over the course of the whole experiment. As one might expect, the vast majority (99.2%) were consistent.

3.1 Results

Did IP addresses respond when their AS path was poisoned? In 64% of cases the answer was “Yes!”. Perhaps more interesting though are the results per AS (we tested multiple IP addresses per AS, and the distribution of the number per AS varied).

We found that 74.8% of ASes $(19,291)$ answered consistently despite the poisoning. The interpretation is that the majority of ASes have a default route configured. Loop detection means that they cannot see the route announcement, and so there is no other standard mechanism for them to reach our prefix.

Of the remaining ASes 20.9% $(5,381)$ did not ever answer, and 4.3% $(1,108)$ answered for some IP addresses, but

[1] Cautionary note: the techniques used in this paper violate the standard convention of the AS-path BGP attribute. While this is not a serious security, or performance problem [4] (we have deliberately kept the rate of such advertisements very low) it does have consequences for other AS topology studies. For instance, our approach will lead to apparently new edges in the AS-graph. Care should be taken in future AS-topology studies to remove the prefix $98.128.0.0/16$ from data taken from this time period.

not others. We noticed that some probes to the test address space failed (e.g., due to "bogon-filters"), but this was a very small percentage, 0.7%. We interpret a non-response as the AS being default-free, though this interpretation is less certain as we cannot say that no IP address in the AS ever uses default.

The mixed result category reveals the complexity of network management practice. For example, we have received one explicit confirmation from an operator who explained that two prefixes were advertised through BGP to their upstream providers, but the rest of the prefixes relied on static routing. He explained that this was because of the IP-TV, and VoIP services. This illustrates that ASes do not have to be homogeneously configured, and we certainly see this clearly in our results.

3.2 The Impact of AS Type

Intuitively, we might expect that ASes that provide transit to other networks will be less likely to use default routes than "stub" ASes. We test this by breaking down our previous results by AS type. However, the naive classification of ASes into transit/stubs by labelling any AS that appears in the middle of an AS path a transit provider and ASes that only originate prefixes a stub, doesn't take into account the business relationships between ASes. Hence we use the classification provided by UCLA [5], which takes into account longer time-periods and additional inference such as customer-provider relationships and node-out-degree to classify the ASes into three classes: stub, small ISP, and large ISP. We use their data from just before our experiment (from April 11, 2009). Our experiment covered $24,224$ (76.9%) of the $31,517$ stub ASes in the UCLA data; $1,307$ (96.0%) of the $1,361$ small ISPs; and 246 (96.5%) of the 255 large ISPs. The UCLA data also contains 8 tier-1 ASes, but our upstream provider uses AS path filtering for paths that contain another tier-1. Therefore, our poisoned announcements are likely to be filtered before these tier-1's, and we therefore don't consider these here.

Table 3 provides the breakdown of our results according to the UCLA classification. The first row of Table 3 gives results for stub ASes: 77.1% have default, 19.3% are default-free, the remaining 3.6% are mixed (including bogons and potential measurement errors). Small ISPs (second row) appear to use default in 44.5% of the cases, 42.2% appear to be default-free, the remaining 13.3% are mixed. In the large ISP category (third row): 17.1% of large ISPs appear to have default, 60.6% appear to be default-free, and 22.3% are mixed. Overall, two trends are noticeable:

1. When going from stubs to large ISPs, the fraction of defaults decreases and the fraction of default-free ISP increases.

2. The number of ASes with mixed results also increases from stub to larger ISPs. This suggests that larger ISPs have different policies and configuration associated with their prefixes. Another factor to keep in mind is that in larger ISPs we typically have more prefixes and measurement points, so that the apparent increasing complexity in their use of default routing is an artifact of the measurement sampling.

	# tested	**default**	**default-free**	**mixed**
stub	24,224	77.1%	19.3%	3.6%
small ISP	1,307	44.5%	42.2%	13.3%
large ISP	246	17.1%	60.6%	22.3%

Figure 3: Fraction of ASes tested with default, default-free, and mixed by category.

The use of default routing is very popular in stub ASes, but less in transit providers, and even less in large transit providers. This is also illustrated by Figure 4, which shows a breakdown of our results (default, default-free, or mixed) against the AS out-degree[2] We see a trend towards decreasing use of default, and increasingly mixed policies as the out-degree increases(at least up to degree 100). About 80% of ASes with out-degree at most 20 rely on default routing. ASes that have high out-degree (300 or more) use default routing in less than 15% of the cases.

The uneven popularity of default routing in different types of ASes is not entirely unexpected, but does complicate the relationship between the data and control planes. For example, if one is initiating a traceroute from a stub network towards some target IP address for which routers on the data path do not have a specific entry, it is possible that the traceroute manages to reach the transit part of the Internet, but stops there. The person initiating the traceroute may then wonder whether there is some problem at the location where the traceroute stops. There is, however, nothing particularly wrong with this location. It is the reachability until the point where the traceroute stops that should be considered surprising from a control-plane viewpoint, since the ISPs in question had no knowledge of the target, and was just using default routing to get to that point. This may cause confusion about the source of a routing problem, and certainly suggests that neither data-plane, nor control-plane measurements are adequate by themselves.

3.3 The Impact of Defaults

The obvious question to ask at this point is "Do defaults matter?" Clearly they matter to the network operators (otherwise why use them so consistently), but how do they impact our measurements? We provide some intuition into this question through simple simulations.

In our simulations, we once again use the AS topology and relationship data provided by UCLA [5]. We could also use our classification of which ASes have default, but this would limit our ability to perform a large number of simulations,

[2] We used a binning of 20 for the x-axis, and all ASes having a degree larger than 300 were put in the 300 bin.

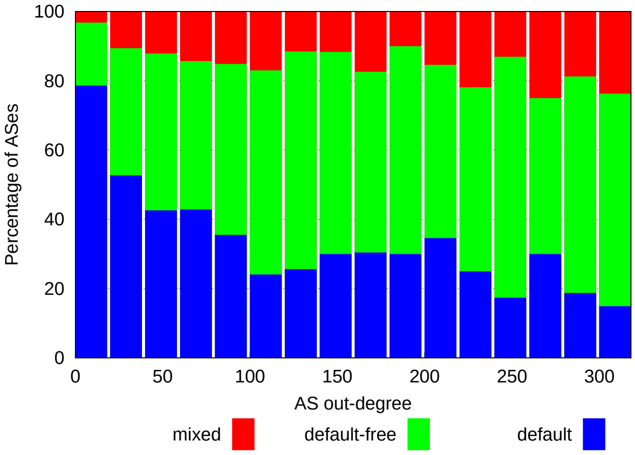

Figure 4: Breakdown of default routing use as a function of AS out-degree.

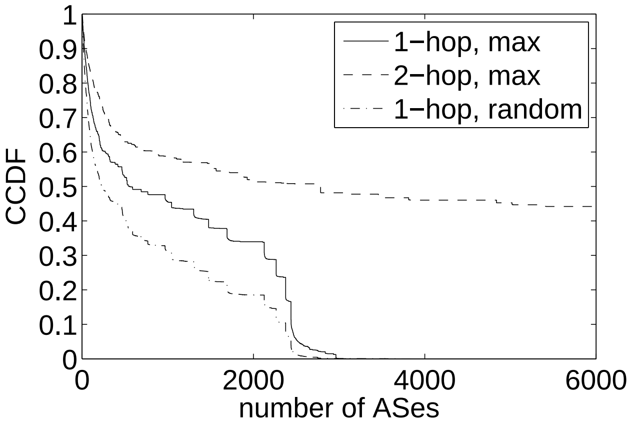

Figure 5: Distribution of ASes reachable using default routes.

and so we use the given topology, but randomly assign which ASes use default, using the probabilities given in Figure 3 (to be conservative, we exclude the mixed cases, and assign default with probability 0.771, 0.445 and 0.171 for stubs, small and large ISPs respectively). For each AS using default, we also need to choose where its default route points. Our measurements don't at present tell us which provider is being used as the default, and so we test two schemes for choosing defaults: random, and max. In the *random* allocation scheme we choose randomly from an AS's providers. In the *max* allocation scheme we choose the provider with the maximum number of customers (i.e., using this measure as a proxy for the size of the provider, we choose the largest provider as the default). These schemes may not be used in practice, but the contrast between them is illuminating.

For each simulated topology we choose a random set of 1000 sources. For each source, we then consider how many potential destination ASes can be reached from this source using default routes. If we use only defaults, then we can reach very few ASes. The Internet hierarchy is relatively flat, so even from a small stub AS, we need only go up through a few layers of providers before we reach a large, or tier-1 provider which doesn't use default. We found that typically only $1-3$ ASes could be reached in this way, and that the maximum was 5.

The more interesting case occurs when we consider an experiment such as our $/25$ advertisement. In this case, the advertisement is accepted by our provider. If we allow that such advertisements will be accepted by immediate providers (but not anyone else), and then consider how many ASes can be reached, we get a distribution of number of ASes as shown in Figure 5 (solid curve). The figure shows the Complementary Cumulative Distribution Function (CCDF) of the number of ASes that can be reached from a random source, given the BGP announcement of destinations propagates one hop (to providers). Note that for the *max* default allocation we can now reach 1000 ASes from approximately 50% of sources, and over 2000 ASes from around $1/3$ of sources. For the *random* allocation of default routes, we can reach somewhat fewer destinations, but the number is still substantial. The contrast between the two allocation schemes is intuitive. By choosing (as our default) an AS with more customers, we make available more potential destinations at each step.

The figure also shows a curve under the assumption that BGP advertisements propagate two-hops. We can see that this has a dramatic impact on the number of ASes that are reachable (6000 are reachable about 50% of the time, with a maximum of nearly $19,000$). For simplicity, we only show this curve for the *max* default allocation, but we see a similar decrease to the 1-hop case when the *random* allocation rule is used.

These simulations obviously over-simplify much of the operation of the Internet. In particular, the propagation of our /25 prefix does not follow a simple "hop-count" mechanism, but is spread depending on the local filters at each AS. However, the simulations do provide us with some valuable intuition. It is quite possible that once the $/25$ reaches our provider, and perhaps a few other ASes that it will be reachable from a significant proportion of the Internet, despite the limited propagation of its routing announcements.

We believe that the experiment of this section also sheds much light on the counter-intuitive results of [3]. The authors of [3] found that the correlation between data-plane and control-plane observations were sometimes surprising, e.g., packets still reached their destination despite the control plane indicating that the corresponding prefix is not reachable. Default routing provides a simple explanation for the phenomena of *reachability without visibility*.

This property of reachability without visibility has not been allowed for in most research on the Internet's topology and routing. Yet it is clear that it provides an unanticipated level of resilience in the Internet's routing architecture, not just at the local ISP level.

4. HIDDEN UPSTREAM DISCOVERY

AS-topologies derived from BGP monitors are known to miss some links [1]. The limited view of BGP monitors leads to missing data, but more importantly there is some evidence that we see most of the customer-provider links, but miss a significant proportion of the peer-to-peer links [6]. This type of measurement bias (if present) is important for topology generation, economic modelling and answering what-if questions. In this section we extend the previous results to test the hypothesis that the customer-provider links are relatively easy to find.

AS-path poisoning was used in the previous section to poison an announcement so that a single AS cannot see it. In this experiment we poison the announcement to all of an AS's upstream providers. If there is a hidden upstream, it will provide an alternative route that will allow connectivity despite our efforts.

Once again we use the AS-topology data from UCLA [7]. This data set uses static BGP snapshots as well as observation from BGP dynamics to determine interconnections between ASes and is therefore considered to be one of the most complete AS-topologies today. In addition, it contains inferred relationships between the ASes from which we may derive a set of upstream providers for each AS.

We need to be able to discriminate between hidden upstreams, and default routes, and so our test set of ASes must be restricted. ASes whose upstream providers use defaults (or have mixed behavior) are unsuitable. In these cases, poisoning upstreams will not necessarily prevent connectivity. We found 966 suitable ASes whose neighbors in the UCLA data do not possess a default route (according to our previous measurements).

We do not need to test each of these ASes separately. For instance, assume AS *x* has neighboring ASes *a*, *b* and *c*, and AS *y* has neighboring ASes *a* and *b*. If we poison *a, b* and *c* we can simultaneously test both AS *x* and *y* for hidden upstream providers. Using this method, we reduced the number of required tests to 406.

Of the 966 test ASes, 912 were not reachable after poisoning their adjacent ASes, implying that they do not have additional upstream providers. Only 33 ASes were still reachable after upstream poisoning, while 5 gave inconsistent results and 16 were suffering from bogon filters (see Section 5.2). So at most 4% of the tested ASes may have upstream provider links that are missing from our sample of the AS topology.

Even these few missing links are in doubt. There are potentially some errors in mapping of IP addresses to AS, so if an IP address *q* was not within the suggested AS *x*, our attempt to poison access to this IP address would fail, and it would appear that additional upstream providers exist. False positives such as this are possible, but false negatives are less likely. They can happen if the link to the hidden upstream is temporarily down during our experiment, or if the hidden provider also uses the same upstreams as the AS itself (it is unlikely that an AS would use such a provider given the limited redundancy it provides). Hence, we regard 33 as an upper bound on the number of missing customer-provider links in our test, in actuality, it is likely that significantly fewer upstream providers are missing.

The results support the belief that standard AS-topology data sees the vast majority of customer-provider links, i.e., BGP sources give very good visibility of upstream providers.

5. TESTING REACHABILITY WITH DUAL PROBING

The experiments described above showed that default routes and hidden connectivity limit control-plane data's ability to predict reachability. Such limitations should be kept in mind before making claims based only on control plane observations. On the other hand, the type of data-plane measurements we have used so far are limited as well. It is easy to find situations where it is hard to interpret the results of ping probes simply because end-host (and middleware) behavior is so varied.

In this section, we describe a rigorous active probing methodology, called *dual probing.* Dual probing makes explicit the assumptions and expectations behind different types of active measurements, and uses them to calibrate expectations.

Before explaining our methodology, we need to better understand the problem of testing reachability. Reachability can be assessed from two different viewpoints:

- How do I see the world?
- How does the world see me?

The first is based on the information a router receives from routing protocols. We addressed the limitations of BGP routing information in previous parts of this paper. The converse question — "how does the rest of the wold see me?" — is something operators often would like to know in order to debug reachability problems. Unfortunately, this information is not directly available from the network layer.

The sampled world viewpoint.

There is data available to see how the world sees us. Services such as BGP monitors, looking glasses, and traceroute servers provide public views of the Internet. However, only a sample of ASes operate a these as a public service, therefore it is hard to get direct data from the *world viewpoint.* What we see when we combine data from the available viewpoints is actually a *sampled world viewpoint*.

A significant problem with this sampled view is that the operators with the sophistication and resources to operate public viewpoints tend to be larger ISPs, nearer the "core" of the Internet [7]. The bias in the viewpoints could mislead. For instance, we might hypothesize that these large, densely connected ISPs have fewer reachability problems than stub ISPs. There is a strong motivation to see a much more complete world viewpoint.

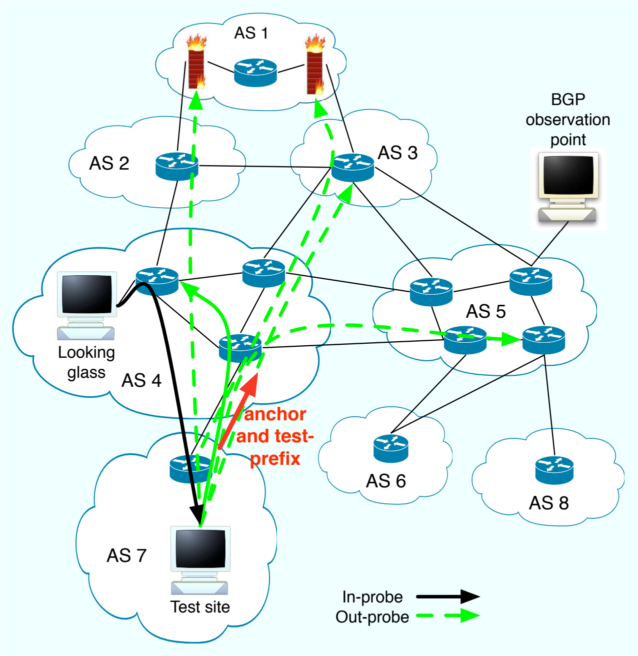

Figure 6: Dual probing: "traditional probing" require looking glass servers. Those are very sparsely distributed and highly biased in what they show. "Out-probes" cover a large fraction of the whole Internet, but, they require that reachability expectations can be calibrated.

Out-probes.

We advocate the use of data-plane probes to provide such a world viewpoint. Suppose a network administrator wants to check that external hosts can reach their network. A simple test would be to ping from a machine towards a large set of external IP addresses covering much of the Internet. If those IP addresses answer the probes, this indicates that the source machine is reachable from the probe's destination. As the units of Internet routing are IP address prefixes, the administrator could assume that reachability exists between this set of prefixes and their site. Obviously there are exceptions, but an administrator is typically not concerned about the end-point connectivity of distant systems. We are concerned that network level reachability exists, such that the end points could in principle connect.

In this section, we term these outbound pings and traceroutes *out-probes*, as probes are sent *out* from the address space to be tested for reachability, even though it is their return that reveals the reachability of our prefix. Figure 6 illustrates this concept. In the "traditional" case (black solid arrow), a probe is in-bound from a public looking glass towards the test-site. In the out-probe case (green dashed arrows), a probe is sent from the test-site towards many destinations in the Internet. Note that the address space under investigation must be the source address of the outgoing IP packet. The probes are aimed outwards to many pingable IPs scattered across the Internet, but it is the return traffic (towards the test address space) that reveals the reachability of the test IP address space.

5.1 Dual-Probe Technique

When the ping probes are not answered, the administrator cannot conclude that their system is unreachable. There are several reasons why outward ping probes might not be answered, among them: (1) the IP address simply does not answer pings, (2) the ping probes are dropped by firewalls on the way towards the probed IP addresses, (3) the IP addresses answer the ping probes but the answers are dropped somewhere on the path back towards the probing host, or (4) there is no route from the IP address in question back to the probing host or vice versa. Only the latter two cases concern reachability of our tester's network. Even case (3) may be a poor indication of unreachability because ICMP probes are often given lower priority and may be dropped preferentially over TCP traffic[3]. So a negative response provides little information by themselves. Useful interpretations can only be obtained when we already know what to expect from a probe.

Calibrating expectations.

If we can calibrate our expectations we will know how to interpret the responses of probes. This is is a similar to what we did in Section 3 where we calibrated our expectations via a look-ahead, i.e., we used two probes separated in time to construct allow for better interpretation of the second probe. Other dimensions we might use for calibration are: probing location, or target address space. We call this approach *dual probing*, though in some cases more than two probes may be involved.

For instance, we can compare probe answers against probes from another prefix, called here an *anchor-prefix*. The anchor-prefix is an old, well-established prefix known to have very good reachability. By comparing the probing results between the *test-prefix* and the anchor-prefix, we have the ability to decide if unanswered pings and traceroutes initiated from the test-prefix are abnormal.

The key behind the success of *dual probing* is the comparison between probes from a test- and anchor-IP. This comparison reveals far more information than a single probe from the test-prefix. Lacking a reply from the anchor probe to a particular IP address we know there is a problem probing this IP address, and so we can discount test measurements as not useful. With a reply to both, we can infer successful reachability. When we receive a ping reply to an anchor probe, but no reply to a test probe we have evidence that there is a reachability problem. The evidence is not conclusive (ICMP packets may be dropped), but over a series of such measurements, we can build confidence in the results.

We also demonstrate that this dual probing approach has a wide dynamic range of applications. It worked well when blockage of the test prefix was sparse, e.g. when used to detect bogon filters (see below), for which there was on the order of 5% blockage. It also worked well at the opposite

[3] Using UDP or TCP probes instead of ICMP does not change the problem as can be seen in Section 6.3.

end of the spectrum, the propagation of a /25 prefix (see Section 2), where visibility was on the order of 5%.

5.2 Bogus bogon filter detection

A *bogon* refers to a bogus routing announcement. These are sent either accidentally, or deliberately to hijack address space, and so ISPs commonly configure either control or data plane filters to prevent traffic to/from obviously bogus addresses. For instance, it is common to refuse traffic or announcements from unallocated addresses space. However, the configuration of these filters does not always keep up as new address space is allocated, so it is important to be able to debug reachability problems caused by bogus bogon filters.

In the past traceroutes from public servers have been used to find these filters [8] (as well as very limited out-probing), but the small sample set of such servers limits the ability to detect bogon filters to a small subsection of the Internet. Here we systematically investigate bogus bogon filters using dual out-probes over a large segment of the whole Internet.

ARIN allocated two large segments of new address space (173.0.0.0/16 and 174.128.0.0/16) for our experiment, and we used 5 smaller segments of this address space. We announced those prefixes from five different locations that volunteered to participate in our experiment: PSGNet in Seattle (USA), Verio in Ashburn (USA), SpaceNet in Munich (Germany), CityLink in Wellington (New Zealand), and IIJ in Tokyo (Japan). Each test-site announced one of the test-prefixes. The anchor-IP was the normal address of a machine inside the ISP that ran the experiment. The test-IP was configured as a secondary IP address on the same interface. We ran three different measurement campaigns to see if bogon filters are removed over time: the first starting in April 14^{th} 2008 ($t1$), the second starting on May 27^{th} 2008 ($t2$), and the third starting on June 12^{th} 2008 ($t3$). About a week was necessary to run all our probes (to the set of target IP addresses described in Section 6.1), so the dates given are approximate. The first measurement campaign occurred before ARIN announced that this address space had been issued. The goal of these measurements was to provide a controlled experiment. We want to understand how prevalent *legitimate* bogon filters are, i.e., how much of the Internet is protected from traffic from unallocated address space.

After the first campaign, the ARIN announced that the address space had been issued, and that it should be removed from bogon black-lists. In addition, we identified (in the first campaign) a list of ASes that definitely filter. The ASes in this list were looked up in the IRR (where possible) and email was sent to operators asking them to (1) confirm our detected filters, and (2) if so to remove them. Thus reachability problems identified in the second set of measurements are genuine problems, which needed to be fixed.

The third set of measurements was used to assess how the reachability problems were changing over time. We will discuss later (Section 6.1) how we chose the list of addresses to ping, and the individual results of those pings, but for the moment let us focus on the problems we discovered. We observed *more than a thousand ASes* that replied to probes sent from the well-established address space, but did not answer probes sent from the test address space. We also saw that the test space showed problems months after it was officially allocated, with little sign that the problem was being corrected! The operator community is aware of this problem but has had no tools to measure the extent of it or to see trends.

In testing for blockage, there are at least two reasons we would wrongly conclude a lack of reachability: (1) ICMP drops and time-outs and (2) upstream filtering. A simple way to compensate for (1) is to repeat probing over time, and from multiple viewpoints, hoping that the concerned hosts or routers do not systematically drop the probes. If we probe one AS several times and it never replies to probes coming from the test address space but consistency replies to probes from the anchor address space, it is likely that this AS does not have reachability to the test space. However, the intent of "debogoning" is that the registries would contact the administrators of incorrectly configured ASes. We do not want to contact the administrators of ASes that are not filtering, as this would degrade the credibility of the service, so false positives must be kept to a minimum. Therefore we define a *filtering likelihood* that takes into account how many return probes we have to the anchor prefix. We do not report the likelihood itself here, but rather break down the observations into several categories.

For an AS to be identified conclusively as having a reachability problem, we require to have zero returns back to the test-IP, and at least five returns to the anchor-IP. If there are zero returns to the test address space, and less than five returns to the anchor IP, we have some indication that the AS might have a problem, but it is less conclusive because of the low sampling. We categorize these as *potential* problems.

The probed AS will not have reachability to the test-prefix if its upstream providers are filtering (and do not use default routing). This type of AS will appear in our list of problematic ASes, but may not be to blame. However, the AS's operator is likely to be interested in knowing that they have limited reachability, and may put upwards pressure on the guilty party to ensure that the problem is rectified. Hence, these identifications, while false, do not degrade the detection service in the same way as other false positives.

It is quite possible that an AS is not configured uniformly, as we saw in the case of default routing. Perhaps filters have been updated on some routers, but not others. Interpreting results from such ASes is much harder. We might try to enter an AS via a working path, or via the blocked path and this may determine our results. Detection is only possible if we have a large number of IP addresses that we probe as well as a large number of probing locations (e.g., [9]).

Figure 7 shows the number of ASes that fall into each category. According to our classification, around 500 ASes are definitely filtering our newly allocated prefix, but near

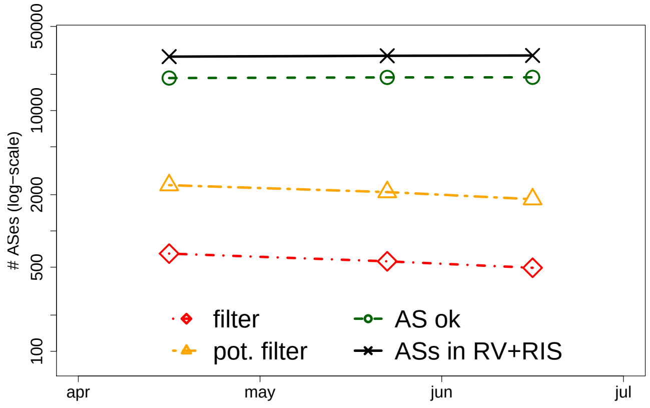

Figure 7: The number of ASes showing conclusive, and probable evidence for reachability problems (log-scale).

2000 are potentially filtering. This means that between 2% and 7% of the whole Internet cannot see our addresses. This is a serious problem! Moreover, the problem hardly changes between our second and third experiment.

To gauge the extent of upstream provider filtering, we considered the location of these reachability problems. Recall that we cannot rigorously determine where probes are dropped, as we do not see the reverse paths[4]. Instead, we studied the nature of the ASes in our list. If a target AS appeared as an intermediate node in any of our traceroute measurements we classify it as a transit AS. If not, then we call it an end-point, or stub-AS. Figure 8 shows the percentage of each category in the ASes with reachability problems. We can see that the vast majority are are stub ASes. This suggests that most problems occur at the edge. This is an intuitively appealing conclusion because it is natural that transit providers – who should have more experience with BGP – are less likely to leave stale filters in their network. The small number (a few hundreds) of transit providers who incorrectly configure filters increases the likelihood that we incorrectly identify an edge node as filtering when it is not.

As Figure 7 shows, bogon filters seem not to be removed quickly. Our experiments over the course of April to June 2008 showed very small changes despite the fact that in mid-April a reminder to operators was sent to remove filters for this address space. This agrees with the operator community perception, and was the reason this experiment was originally commissioned.

Feedback from network administrators.

The first campaign identified a list of ASes that had no connectivity to our address space. We used the Internet Routing Registries (IRRs) to find e-mail contacts for these ASes. We contacted around 75 operators manually via e-mail, asking them if they were filtering the newly allocated address space. We only received 17 replies. The majority, ten ISPs, confirmed that they had out-dated bogon filters. Two told us it was their upstream that was filtering. We see those as encouraging responses, as our methodology is about finding places with no reachability, and not about “blaming” operators. This shows again, how careful we have to be in using the methodology, as our methodology can detect regions where limited connectivity exists, but should not be used for finger-pointing. Therefore, implementing such a service within the registries has to be considered with care, as for example ASes with default routes pointing to their upstream are affected by the filters of their upstream.

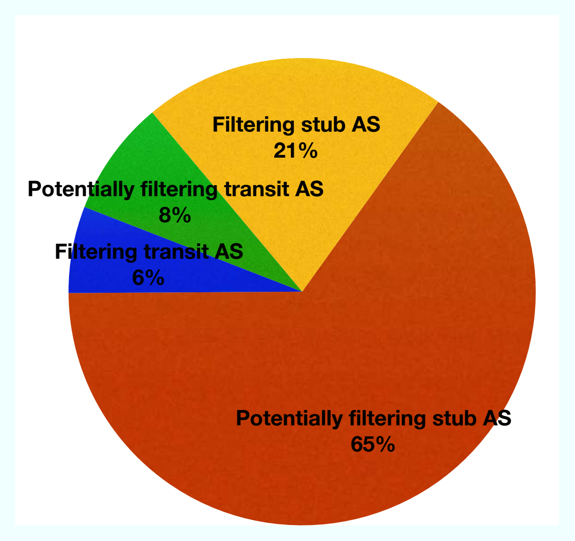

Figure 8: Percentage of ASes with reachability problems based on transit/non-transit classification.

In addition, we got three very confused responses from operators asking us what we are talking about. This lack of understanding of the the issue suggested that these ASes were probably operated by folk who don’t understand how maintain their filters, and although the responses neither confirmed not denied the existence of bogon-filters, they certainly left us suspecting the filters were there.

Further, however, we received two replies saying they had no such filters. For one of those, we had not contacted the ISP itself but rather, due to an IRR look-up error, the APNIC helpdesk, which is obviously not filtering. The final one did not mention whether or not connectivity existed, so they might also fall in the category where the upstream is filtering.

6. IMPACT OF METHODOLOGICAL ISSUES ON MEASUREMENT CONFIDENCE

So far we have shown, how observations from the control plane suffer from “visibility issues”, and that the data plane can offer a different perspective. Obviously, data-plane observations have their own limitations. In this section we discuss three methodological issues that have to be understood to ensure the success of an experiment aimed at making con-

[4] Relying on a tool such as [2] may partly solve this problem.

clusions about reachability from active probing: the topological coverage of the probes (Section 6.1), the mapping from IP address to AS number (Section 6.2), and finally the type of probes used (Section 6.3).

6.1 Topological coverage

The motivation for out-probes (see Section 5.1) is to "look" into those areas of the Internet where no BGP monitors or looking glasses exist, i.e. mainly the edge. Unfortunately, this part of the Internet is large, and changing quickly [7]. Our visibility of the edge is not as good as in the core. The idea is to measure reachability by sending outbound probes to "the edge" of the Internet and draw conclusions based on the responses. We have already discussed in Section 5.1 the need to calibrate our expectations, to be able to draw conclusions. While the calibration will tell us whether or not to expect an answer in a particular case, we need a list of IP addresses that we can expect to answer our probes.

Several requirements for the IP address list need to be considered, and some of them conflict with each other. First we would like to have a wide coverage, e.g., reach as many ASes in the Internet as possible. Second, we would like to probe inside ASes using a fine granularity to see non-homogeneously configured patterns. Third, we also like to limit the number of probes that we have to send and the time it takes to probe all IP addresses.

In this section we discuss the properties of the list of *pingable* IP addresses we used. ASes are often not homogeneously configured [10], a per AS-granularity might not be sufficient. However, the number of ASes is in the order of tens-of-thousands, while the number of prefixes is an order of magnitude larger. At the edge of the Internet, it is particularly difficult to find enough pingable IPs. Besides the problem of the probing granularity, we also need a large enough number of measurements to build confidence in the results[5].

Note that as our goal is not to do topology discovery per se, we did not try to achieve a proper coverage of the router-level topology as done by topology discovery projects [11–14], and neither do we seek the level of detailed coverage of [15], as we are not concerned with the behavior of the end-hosts (which after all may change minute by minute), simply the reachability of the end systems to which they connect.

6.1.1 *Finding pingable IP addresses*

Obtaining a large number of pingable IP addresses is one of the important issues for large-scale topology discovery [11–14, 16, 17]. Many projects use existing lists of IP addresses such as the one from CAIDA [11]. This list is based on IP addresses observed from passive measurements: packet capture, DNS requests observed at root servers, and Web servers logs. However, we also used active discovery techniques such as developed by Sugih et. al [18], as well as brute-force scanning for a very small portion of the address space where required. Together with sharing and merging our IPs lists with that of other researchers we obtained a pool of $4,655,238$ IP addresses in total.

In general we have to decide what granularity to aim for: router-level or AS-level. Depending on the application this might vary. The router-level topology is more detailed, but harder to obtain with any certainty. Moreover, for bogon-filter discovery the aim might be to contact network operators to remove filters. In such cases, we just have to assure to have high enough chance to discover non-homogeneous filters, but contacting the ISP will be done on a AS-level granularity.

6.1.2 *Coverage at the AS-level*

We selected a subset of $306,780$ IP addresses from our pool of IP addresses that we use for our measurements. The resulting coverage was of $154,683$ pingable prefixes in $25,780$ pingable ASes. We selected those IP addresses based on the following objectives:

- Probe as many ASes as possible.
- Aim ideally at 30 pingable IP addresses per AS, unless there is a reason to believe that a finer granularity within that AS is required. This number of 30 IP addresses is an arbitrary threshold: it should be large enough to allow some estimations about the required granularity, while reducing the number of probes needed. Note that if far more than 30 pingable IP addresses are available inside an AS, we will limit their number to 30 to prevent spending effort on ASes where we can too easily find pingable IP addresses.
- Aim at covering as many diverse prefixes within an AS as possible, e.g., take those 30 pingable IP addresses from as many prefixes as available inside that AS.
- If all prefixes are covered, and still our threshold of 30 is not reached, then improve confidence, by obtaining several pingable IP addresses within the considered prefix.
- Minimize the number of probes sent.

We wanted to keep both the number of probes sent as well as the time necessary to run a probing campaign reasonably low. Therefore, we chose to limit the number of IP addresses probed. As the underlying Internet topology may change while we are probing, taking more time to run our probing may lower the quality of our results. Even recent probing tools, such as Paris-traceroute [19], are too slow to probe a very large list of IP addresses in a reasonable amount of time. On the other hand, probing too fast may also not be desirable, as many routers are known to be configured to rate-limit the number of ICMP packets [20]. A slower probing rate might also be required to avoid many packet drops or having the probing host black-listed.

[5] There are several reasons why we want to have multiple measurements within an AS/prefix: For example to compensate for measurement errors, such as packet loss. To deal with mapping errors, such as a customer administrated router which is provisioned on one interface with provider IP address space. For us this router would appear as belonging to the provider administration.

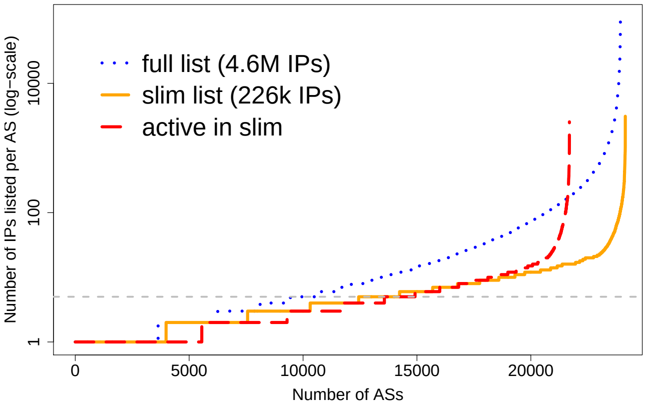

Figure 9: Total number of pingable IP addresses per AS: stubs (black, top curve), small ISPs (green, middle curve), and large ISPs (blue, lower curve). ASes with more than 300 IPs fall in the last value.

Figure 9 shows a CDF the number of IP addresses that we have on a per AS basis. The x-axis shows the number of pingable IP addresses we have per AS, and on the y-axis we plot the fraction of ASes for which we have less than *x* pingable IP addresses. The solid black curve shows stub ASes, the green dots show small ISPs and the blue dots show large ISPs (see [5] for the classification). We also show as a grey line the value of 30 pingable IP addresses. Finding pingable IP addresses at the edge in each AS is difficult. 86.6% of stub ASes ($20,980$ out of $24,224$) do not reach this threshold. Actually, for 31.3% ($7,589$) of the stub ASes we have only one pingable IP address. For small and large ISPs on the other hand, traceroutes easily sample many IPs within the network core. Most ASes do reach the threshold of 30 pingable IP addresses: 63.7% of small ISPs (833 out of 1307) and 82.1% of large ISPs (202 out of 246).

6.1.3 *Stability of pingable IP addresses*

Depending on the quality of our list of pingable IP addresses, different regions of the Internet may not be covered as well as others, or too few pingable IP addresses will render the results inconclusive. Once an initial list is built, we must maintain its coverage over time. IP addresses may belong to hosts whose connectivity to the Internet changes over time, or may be dynamically allocated to different hosts. For example, consider "dial-up" IPs. They may respond at some time, and not a few hours later. Even though we calibrate our expectations while we run our tests, we still rely to some degree on the expectation that IP addresses are stable and give predictable responses. We have to continuously monitor which IP addresses are pingable and drop from our list IP addresses that are chronically inaccessible. We have to detect when our coverage of an AS becomes low, and then we have to add IP addresses to the list to compensate for this inadequate coverage.

For a methodology that seeks a given coverage of the Internet to be successful, it is important to have a good understanding of the quality of the list of pingable IP addresses on which it relies. Ideally, this should be a service offered from the route registries, where operators could register IPs that are responsive to pings and could be used to determine reachability. In this section we study the changes in the availability of our IP addresses over time, as well as the sensitivity of results to different probing locations.

We found that our list of pingable IP addresses is fairly stable. In April until June 2008 we evaluated the stability of our IP address list. We observed that 95.8% of all IPs that were pingable in April remained pingable during the following two month. However, 2.2% of those IP addresses did not respond to our pings in those two months. CAIDA [11] reports a decay rate of their list of active IP addresses of about 2 to 3% per month.

As usual when working with measurement data, a certain fraction of IP addresses behave strangely. In our case these may be artifacts induced by the availability of end-hosts, which might sometimes be up, sometimes not (e.g., dial-ups). Another cause might be ICMP related measurement problems due to packet drops (see Section 6.3 for more details). A recent study [20] found that routers tend to increasingly drop direct probes. Indirect probes on the other hand (e.g. traceroutes) do not seem to be concerned by this trend. Some routers, often at the edge, react to probing with ICMP rate-limiting techniques. Overloaded links also may drop probe packets more frequently than other packets. Furthermore, some hosts respond very slowly to probes which create time-out issues for the probing tools. If we sum up all those artifacts, we estimate our error roughly between 2% and 5%.

6.1.4 *Consistency across probing locations*

When assessing reachability, the location from which the address space is advertised (the test-box) may yield different results. Take for example the "Rocketfuel" work [9], aimed at discovering the internal topology of ISPs. To achieve this, the authors probed from different locations towards different IPs. Non-homogeneously configured routers or packet filters only configured on certain links may impact the results of our probing. Depending on how our probes enter a given AS and traverses it, we may or may not sample bogon filtering routers or show differences in our default-route experiment. It is therefore reasonable to assume that differences between locations may exist.

However, the results of our bogon-experiments (section 5.2) do not suggest a very large influence of the probing location. We set up two test-boxes in the US, one located in Seattle on the West Coast, and one in Ashburn on the East Coast. We had another test-box in Tokyo, Japan and one located in Wellington, New Zealand. Comparing the results from those locations, we found that in 93.75% of the cases, the location did not matter at all.

We compared the three measurement campaigns and only for 0.55% of the cases, we observed a difference between the locations during all three campaigns. This is not a very significant proportion of our pingable IP addresses, especially in the light of measurement artifacts, e.g. inaccurate IP to AS mapping or ICMP related measurement artifacts. We observed differences between the locations during one campaign in 4.2% of the cases and during two campaigns in 1.5% of the cases. Given our results, we cannot claim that there are significant differences between locations. Making confident inferences about why differences are observed in reachability is very difficult without additional information.

So far we talked about differences between locations on an IP address basis, but many filtering policies are implemented AS-wide. For instance, network administrators typically place bogon filters at all their border routers. While these filters affect certain prefixes, we are looking for the ASes that are configuring those filters, not their victims. Similarly, router misconfigurations [21] and routing instabilities [22] often exhibit large differences on a per-AS basis. Many reachability problems can be seen at the AS granularity, without delving into individual IPs.

In this case it is good to have several IP addresses to probe within an AS to compare their results. We noticed that for 84.9% of all probed ASes, all IPs within those ASes show no differences between the probing locations. If there are differences, this typically affects only very few IPs out of the set of probed IPs within that AS. This suggests that measurement artifacts are more likely to be the cause of inconsistencies between probing sites than differences in reachability.

6.2 IP to AS number mapping

A general issue we encountered during all experiments is the mapping of IP to AS number, e.g., [23–26]. To determine the AS number corresponding to an IP address, one performs a lookup in BGP routing tables. However, it is common that customer routers get upstream-facing IP addresses from their provider's address space. Thus, when a probe enters the customer AS, the response comes from an IP address that still belongs to the provider's address space, which is managed by the customer. In our dual probing for example (Section 5.2), the ping initiated from the anchor prefix would indicate, incorrectly, that the customer router belongs to the provider's AS. If the customer does not respond to the test-prefix probe, but their upstream provider would, we would mis-diagnose that the provider's IP address did not respond.

Not only does the IP to AS mapping create issues such as the one mentioned before, but changes in the advertised address space require us to monitor changes in the mappings. We observed the changes in mapping over time during the default routing experiment (Section 3). We first performed an IP to AS mapping of our IP address list in 2007, and compared it with another mapping based on a set of BGP routing tables from RIPE and Routeviews (see Section 3) dating from the same period as the default routing experiment (2009). We noticed that on a per prefix basis things changed quite a bit from 2007 to 2009. Only 88.0% of the prefixes still had the same mapping in 2009. 7.43% of the prefixes were not in the BGP routing tables anymore. By this we mean exact matches of a given prefix. If for some reason the prefix was not visible in RIPE or Routeviews but only a more or less specific prefix was in the table on that day, it appears as if the prefix is not in the routing table. In less than 0.5% of the cases, a prefix showed multiple origin ASes (MOAS). MOAS are problematic as we typically do not necessarily know on which AS number they react to.

IP to AS number mapping can also be a general problem in the default route methodology. Imagine we probe IP_1, which we have mapped to AS 1, but which would actually belong within AS 2. If we do AS path poisoning with AS 1, IP_1 would reply (as its administratively with AS 2) and therefore we mis-classify AS 1 as having default.

6.3 What type of probe to use?

When designing an experiment based on active probing, one of the basic questions one has to answer is which probing tool to use, e.g. ping or traceroute? In theory, ping should be sufficient if one is primarily concerned to know if the probed address space has reachability towards the tested address space. In case reachability cannot be established, neither ping nor traceroute will give the information we want most – the return path taken by the probes. Predicting this return path can be done only very partially by leveraging the record-route option and using forward probing from different locations [2]. The limited number of locations from which probing can be done in practice implies that guessing the reverse path of traceroutes will typically be a highly underconstrained inference problem. Only a tool that will record the full path of the probe and its answer will solve this limitation.

ICMP vs. UDP.

When relying on ping probes, the type of packet (ICMP, UDP, or TCP) used affects significantly the chance of obtaining an answer.

We observed the responsiveness of both ICMP and UDP probes during the three measurement campaigns of the bogon filtering experiment (Section 5.2). We found similar results across the three campaigns. For ICMP, roughly 70% of the IP addresses were reachable. For UDP, the responsiveness was only 30%. By comparing those unsuccessful UDP probes with the corresponding ICMP probe that reached the destination, we observed that 74% of those UDP probes stopped one hop before the destination. Firewalls thus tend to filter UDP even when ICMP pings go through. The firewall is usually not on the destination host itself, but rather at the site edge, which would be one or two IP-hops before the destination host. We observed that 90% of those UDP probes got filtered either at the destination hop or exactly one IP-hop before, and 98% got filtered less than two IP-hops away from the the destination. We also tried TCP, but the responsiveness of our probes was even worse than UDP – around 5%, likely for similar reasons.

We confirm the results from [20] that also found ICMP probes superior to UDP and TCP. However, the responsiveness of the IP addresses we probed is quite different from what is reported in [20]. The difference between our results and those from [20] probably lies in the type of hosts probed. We chose our pingable IP addresses in such a way that we cover as many ASes in the Internet. [20] were targeting specifically the responsiveness of routers and did not aim at a wide coverage of ASes.

7. RELATED WORK

This work addresses one of the most fundamental services of the Internet: reachability. It is therefore not surprising to see numerous papers and presentations at conferences and workshops. Some of those studies are interested in how "happy" the packets are [3, 27]. Most recent works have studied pathological behaviors related to the address space, e.g. bogon advertisements [8, 28], hijacking [29, 30], misconfigurations [31], or DDoS attacks [32].

However, most research studies have so far concentrated on BGP [33]. Slow BGP convergence [34, 35], issues with policy routing [36], oscillations in BGP [37], and routing instabilities [22] are among the many problems encountered.

Few have looked at the data plane as a means of exploring the control plane, though [38] explores some issues of prefix propagation similar to our testing of a /25 prefix.

Researchers and practitioners either have to gather their own data or rely upon data collected by various sources, such as CAIDA's Skitter successor, Archipelago [11], and large BGP data collection projects like RIPE Routing Information Service [39] or Oregon RouteViews [40], or Team Cymru's services [26]. Available datasets have such partial visibility of the Internet topology [41, 42] that it is unwise to rely on them to debug reachability.

Few tools are available to debug reachability, *ping* and *traceroute* are the most widely used today. Those tools suffer from important limitations, which are hardly compensated by trying to obtain part of the reverse path [2]. Ping can be filtered by firewalls and NATs. In addition, ping probes between any two hosts, cannot confirm that reachability is bug-free. If some routers do not propagate part of the address space, routing will try to find paths around the problematic regions. In that case, routing will use a "suboptimal path". Often this is not considered when we are concerned with simple reachability, but suboptimal routing may result in intermittent problems as packets are rerouted and these are very hard to debug by normal means.

Traceroute reveals paths, but also has many limitations. Dynamics in the routing paths or load-balancing are only partially visible by traceroute [19]. The record route option in IP packets [43, 44] provides another way to sample the forward path from a host towards a probed host though it is not always supported, and has hop distance limitations [14]. Most importantly for this study no technique known today measures the return path from the probed host towards the probing host. The reverse path would give valuable information about the suboptimal routing, but we are restricted (in finding this information) to public traceroute servers that can create a forward path towards our test equipment.

8. CONCLUSION AND FUTURE WORK

In this paper we discussed the biases and practices that make reachability more complex than it appears from publicly available BGP data. Our experiments show that data-plane reachability is different from control plane visibility. Features such as default routing are common, and provide alternative means for packets to reach their destination even when a route has failed to propagate widely.

We have also explained the need for improved methodologies when debugging reachability issues. We showed that because of the limitations of current probing tools, building methodologies to debug reachability issues need far more care and effort than might be expected by the networking community, but we present two techniques that have been very useful here: route poisoning, and dual probing.

Our work clearly supports more work towards better assessments of data plane and control plane performance and their interactions. For instance, we believe that our results shed light on unexpected results about data plane behavior [3] that revealed non-trivial relationships between the data plane and the control plane.

Acknowledgments

We are grateful to ARIN for loaning previously unallocated prefixes to us. In addition we thank Gert Doering and SpaceNet, Andy Linton and Citylink, Matsuzaki Yoshinobu and IIJ for their efforts in announcing the test and anchor address space and doing all the test runs for us. This work was supported by a gift from Cisco, and S. Uhlig was partially supported by G-lab (`http://www.german-lab.de`), a research project (support code 01 BK 0805) funded by the Federal Ministry of Education and Research of the Federal Republic of Germany. M.Roughan was in part supported by The Australian Research Council through grant DP0665427.

9. REFERENCES

[1] 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 *Proceedings of ACM SIGMETRICS*, (Annapolis, USA), June 2008.

[2] E. Katz-Bassett, H. Madhyastha, V. Adhikari, A. Krishnamurthy, and T. Anderson, "Reverse traceroute," 2009.

[3] F. Wang, Z. M. Mao, J. Wang, L. Gao, and R. Bush, "A Measurement Study on the Impact of Routing Events on End-to-End Internet Path Performance," in *Proc. ACM SIGCOMM*, 2006.

[4] L. Colitti, *Internet Topology Discovery Using Active Probing*. PhD thesis, University di "Roma Tre", 2006.

[5] R. Oliveira and B. Zhang, "IRL - Internet Topology Collection," 2009. `http://irl.cs.ucla.edu/topology/`.

[6] M. Roughan, S. J. Tuke, and O. Maennel, "Bigfoot, sasquatch, the yeti and other missing links: what we don't know about the AS

graph," in *IMC '08: Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, (New York, NY, USA), pp. 325–330, ACM, 2008.

[7] R. Oliveira, B. Zhang, and L. Zhang, "Observing the Evolution of Internet AS Topology," in *Proceedings of ACM SIGCOMM*, August 2007.

[8] N. Feamster, J. Jung, and H. Balakrishnan, "An empirical study of bogon route advertisements," *ACM Comput. Commun. Rev.*, vol. 35, no. 1, pp. 63–70, 2005.

[9] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP Topologies with Rocketfuel," in *Proc. ACM SIGCOMM*, 2002.

[10] M. Caesar and J. Rexford, "BGP routing policies in ISP networks," *IEEE Network*, vol. 19, no. 6, 2005.

[11] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet mapping: From art to science," in *Proc. of IEEE Conference For Homeland Security, Cybersecurity Applications & Technology*, 2009.

[12] R. Govindan and H. Tangmunarunkit, "Heuristics for Internet map discovery," in *Proc. IEEE INFOCOM*, Computer Science Department, University of Southern California, 2000.

[13] Y. Shavitt and E. Shir, "DIMES: let the Internet measure itself," *ACM CCR*, vol. 35, no. 5, pp. 71–74, 2005.

[14] R. Sherwood, A. Bender, and N. Spring, "Discarte: a disjunctive Internet cartographer," in *Proceedings of ACM SIGCOMM*, 2008.

[15] J. Heidemann, Y. Pradkin, R. Govindan, C. Papadopoulos, G. Bartlett, and J. Bannister, "Census and survey of the visible Internet," in *IMC '08: Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, (New York, NY, USA), pp. 169–182, ACM, 2008.

[16] B. Donnet, P. Raoult, T. Friedman, and M. Crovella, "Efficient algorithms for large-scale topology discovery," in *Proc. of ACM SIGMETRICS*, pp. 327–338, 2005.

[17] B. Donnet and T. Friedman, "Internet topology discovery: a survey," *IEEE Communications Surveys and Tutorials*, vol. 9, pp. 2–15, December 2007.

[18] A. Z. Sugih and S. Jamin, "Rapid exploration of Internet live address space using optimal discovery path," in *In Proc. Global Communications Conference*, 2003.

[19] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, "Avoiding traceroute anomalies with Paris traceroute," in *Proceedings of the 6th ACM SIGCOMM on Internet measurement*, 2006.

[20] M. H. Gunes and K. Saraç, "Analyzing router responsiveness to active measurement probes," in *Proc. of Passive and Active Measurement conference (PAM)*, 2009.

[21] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP Misconfigurations," in *Proc. ACM SIGCOMM*, September 2002.

[22] A. Feldmann, O. Maennel, Z. Mao, A. Berger, and B. Maggs, "Locating Internet routing instabilities," in *Proc. ACM SIGCOMM*, (Portland, OR), pp. 205–218, 2004.

[23] Z. M. Mao, J. Rexford, J. Wang, and R. H. Katz, "Towards an accurate AS-level traceroute tool," in *Proc. ACM SIGCOMM*, (New York, NY, USA), pp. 365–378, ACM, 2003.

[24] H. Chang, S. Jamin, and W. Willinger, "Inferring AS-level Internet topology from router-level path traces," in *In Proceedings of SPIE ITCom 2001*, 2001.

[25] Y. H. Andre, Y. Hyun, A. Broido, and K. Claffy, "On third-party addresses in traceroute paths," 2003.

[26] Team Cymru. `http://www.team-cymru.org/`.

[27] V. Paxson, "End-to-end routing behavior in the Internet," *IEEE/ACM Trans. Networking*, 1997.

[28] C. Labovitz and A. Ahuja, "Shining Light on Dark Internet Address Space," *NANOG 23*, 2001.

[29] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang, "PHAS: A prefix hijack alert system," in *15th USENIX Security Symp.*, 2006.

[30] P. Boothe, J. Hiebert, and R. Bush, "How Prevalent is Prefix Hijacking on the Internet?," *NANOG 36*, February 2006.

[31] D. Wetherall, R. Mahajan, and T. Anderson, "Understanding BGP misconfigurations," in *Proc. ACM SIGCOMM*, 2002.

[32] Y. Chen, A. Bargteil, D. Bindel, R. Katz, and J. Kubiatowicz, "Quantifying network denial of service: A location service case study," in *ICICS '01: Proceedings of the Third International Conference on Information and Communications Security*, 2001.

[33] T. G. Griffin Interdomain routing links. `http://www.cl.cam.ac.uk/users/tgg22/interdomain/`.

[34] C. Labovitz, R. Malan, and F. Jahanian, "Internet routing instability," *IEEE/ACM Trans. Networking*, 1998.

[35] Z. M. Mao, R. Govindan, G. Varghese, and R. Katz, "Route flap damping exacerbates Internet routing convergence," in *Proc. ACM SIGCOMM*, 2002.

[36] T. G. Griffin and G. Huston, "BGP Wedgies," 2005. RFC 4264.

[37] D. McPerson, V. Gill, D. Walton, and A. Retana, "Border Gateway Protocol (BGP) Persistent Route Oscillation Condition," August 2002. Internet Engineering Task Force, RFC 3345.

[38] R. Beverly and S. Bauer, "The spoofer project: Inferring the extent of source address filtering on the internet," in *Proceedings of USENIX Steps to Reducing Unwanted Traffic on the Internet (SRUTI) Workshop*, pp. 53–59, July 2005.

[39] RIPE's Routing Information Service. `http://www.ripe.net/ris/`.

[40] University of Oregon RouteViews project. `http://www.routeviews.org/`.

[41] A. Lakhina, J. Byers, M. Crovella, and P. Xie, "Sampling biases in IP topology measurements," in *Proc. IEEE INFOCOM*, (San Francisco, CA), April 2003.

[42] H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Towards capturing representative AS-level Internet topologies," *Computer Networks Journal*, vol. 44, April 2004.

[43] J. Postel, "Internet protocol specification." Internet Engineering Task Force, RFC791, September 1981.

[44] F. Baker, "Requirements for IP version 4 routers." Internet Engineering Task Force, RFC1812, June 1995.