BGP Prefix Hijack: An Empirical Investigation of a Theoretical Effect

Masters Project

Advisor: Sharon Goldberg

Adam Udi

Introduction

Interdomain routing, the primary method of communication on the internet, is known to be vulnerable to class of attack called prefix hijack. Several studies [10] [9] have made claims that state BGP prefix hijack attacks are more effective on tier 1 (T1) origins than tier 2 (T2) or stub origins. However, to our knowledge all of the claims have been based on modeling studies, without empirical evidence. Thus, the point of this study is to investigate the hypothesis using empirical data from an actual prefix hijack event.

On December 24th, 2004, AS9121 (TTNET Turk Telekomunikasyon Anonim Sirketi) started announcing ownership of well over 100,000 prefixes. This was a true break from form, as in the week leading up to 12/24/2004 AS9121 announced about 100 prefixes. Unfortunately for most internet users, many large networks believed at least some of the routes, with AS6762 (Telecom Italia Seabone) believing nearly all of them. The end result was that for most end users, huge chunks of the internet were unreachable for at least several hours. [11]

We find no evidence to prove or disprove the claim that T1 origins are the most vulnerable. We believe that our results are inconclusive for two reasons: first, the attacking AS, AS9121, is a T2 AS which could (TODO: FIX) mean any attack it launches is too effective [10], drowning out the effect we expect to see. Second, and more importantly, we don't believe that the data we analyzed has enough visibility into the network to make any strong conclusion. For instance, we were only able to directly observe hundreds of ASes falling for this attack, even though there are about 40,000 ASes in the internet.

Background

The internet is comprised of thousands of autonomous systems (ASes) networked together to form a giant graph. At its core, there are between 11 and 15 large Tier 1 ASes which are highly connected. Traveling outward are the Tier 2 ASes, which are typically strongly connected, but not quite as large as Tier 1 ASes. Finally, most ASes are stub ASes, typically only connecting to one or two T1 or T2 ASes - they are the "leaves" in the graph.

A typical packet travels through several ASes before it reaches its destination AS. Smaller ASes almost always pay larger ASes for service. That is, a small AS pays for all traffic which it wants to send through a larger AS, and for all traffic that the larger AS carries through to itself. Some peering arrangements occur, specifically all the T1 ASes carry traffic for each other free of charge.

Border Gateway Protocol (BGP) is the protocol which governs how ASes inform each other about connectivity. Each AS sends and receives messages all day, notifying each other of their current routes to specific prefixes. For each prefix, an AS is typically faced with a set of routes to choose from.

The classical economic model ([10] [4] [5] [2] [3]) states that ASes generally follow the following set of rules when choosing a route:

1. Separate all routes which would earn money and choose the shortest
2. If no routes earn money, choose the shortest route which doesn't cost money
3. If no routes are free, then choose the shortest route you know.

In BGP when an AS receives a BGP update it has no way of verifying the information it contains. As a result, AS routers nearly always accept the contents of BGP updates as fact. All this means that an AS can lie about which IP prefixes it owns and draw traffic to itself. Such an event is called a BGP prefix hijack attack, and several have been documented over the last decade with varying amounts of success.

This project is about understanding which type of AS is most vulnerable to prefix hijack. At first glance the routing rules seem to offer up an intuitive solution: IP addresses originating in stub ASes will typically already generate money, so an attacker will have more trouble convincing other ASes to route to him instead of the true origin. In contrast, a larger (T1) AS will be associated with high cost, so it should be easier for an attacker to draw traffic away from the bigger, "more powerful" ASes. [10]

Data Collection

The first step was to collect all the data we could from Ripe [6] and Routeviews [8] in the week leading up to the attack. We use BGP updates as our primary data source.

We also use UCLA's Cyclops [1] data set as our AS graph. This data was leveraged exclusively to determine AS size (how many neighbors it has).

Finally, we use the Routing Report [7] data to determine prefix ownership. Another way we might have done this was to look at BGP updates and routing tables to determine ownership.

Terminology

We use the following terms in this paper.

1. AS: Autonomous System
2. Attacking AS: AS9121 - the AS which launched the prefix hijack
3. Origin AS: The AS who is the true origin (owner) of a prefix which was hijacked by AS9121
4. Tricked AS (or victim AS): an AS who was routing to the attacking AS instead of the Origin AS
5. Attacked prefix: A prefix not belonging to AS9121 which was announced by AS9121 on 12/24/2004.

Process and Results

Our first step was to try and discover the set of attacked prefixes. To do this, we determined which prefixes we saw leading to the attacking AS on 12/24 (about 105,000 total). Next, we simply subtracted out every prefix which we saw going to the attacking AS in the week leading up to the attack (only about 100 or so).

Our next step was to collect every BGP announcement which included any of the attacked prefixes and begin analysis. The first metric we looked at was prefix vs number of tricked ASes (see Figure 1). See appendix Figure 5 to see the same data split by origin size. The most striking thing about the figure was not the shape of the graph, it was that for most prefixes, our data we only observe 20 or less ASes being tricked. Even more striking, at the highest end we only see about 150 ASes tricked, a highly unlikely number considering the reported scale of the hijack.

Since we weren't seeing nearly as many ASes as we expected, we decided to extrapolate who else might have been tricked using an AS graph[1]. We inflated

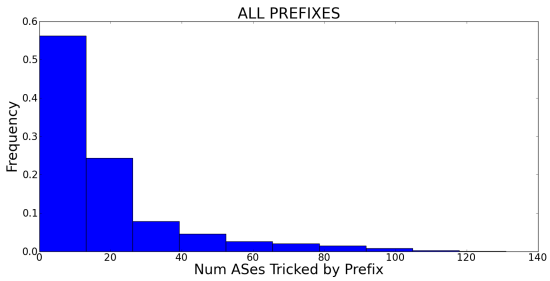


Figure 1: Histogram of Number of ASes tricked (by prefix)

the number of tricked ASes using the following metric: After generating the list of tricked ASes from the BGP update data, we inspect the set of neighbors of every AS and added any it to the tricked set if all of its neighbors were also tricked. For example, any stub with a single provider is classified as tricked if its provider is tricked. We believe this to be a conservative metric, giving us a strict lower bound on tricked ASes. Figure 2 shows the results. Note: we see as many as 4000 ASes tricked in this graph. See appendix Figures 7, 8 to see the results split up by origin size.

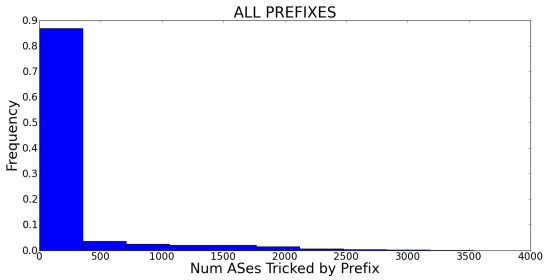


Figure 2: Histogram of Number of ASes tricked (by prefix, extrapolated)

As we didn't see any clear trends by analyzing the origin size, the next angle we tackled the problem from was origin distance. See appendix Figures 10, 11 for the results. In summary, we discover that the distance the origin is from the attacker appears to have no affect on how widespread the attack is.

Finally, we tried one more method. We attempted to quantify whether any particular AS being tricked meant that the attack would be largely successful. Figure 3 shows AS vs (number of routes that AS shows up in / number of prefixes in group). Each data point roughly represents how often a specific AS

shows up in tricked routes, with prefixes aggregated by the size of the tricked set. One possible bias in the data is that ASes closer to the attacker are most likely to show up in routes. So, Figure 4 shows the same plot, but only includes ASes which are exactly one hop from the attacker.

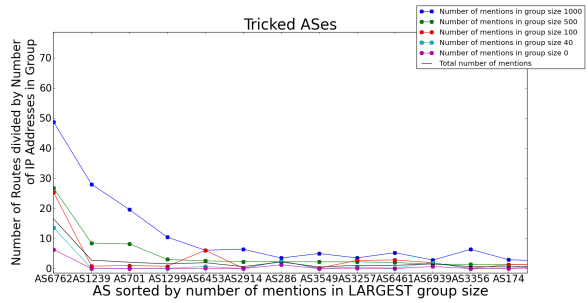


Figure 3: AS vs Number of Routes

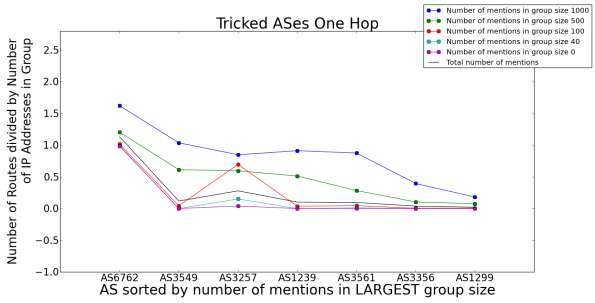


Figure 4: AS vs Number of Routes (one hop)

Figure 4 says something interesting - if AS6762 was tricked, it is much more likely that many ASes would be tricked as well.

See the appendix to see everything we tried to analyze the data.

Conclusions

During the course of this study we looked at the data from many angles. The evidence we provide in this paper reveals very little. As a result, we are unable to prove or disprove the original hypothesis. We believe we were unable to conclude anything not because nothing is there, but because of two obstructing factors.

First, AS9121 is a large T2 AS with many connections. Initially we thought this would help, as we would have more data to look in to. However, we observe that this is not the case. We believe that we see an effect mentioned in [10], that T2 ASes can launch the most successful attacks. It is possible that this effect eclipses the effect we hoped to see.

The second and more problematic factor is one of vantage point. Our data comes from several (10-12) collectors which listen to all the traffic they can hear on a link between a pair of ASes. Obviously, the internet graph is much larger than 12 edges, so we aren't able to see the lions share of updates that travel the internet. Even worse, only the biggest ASes actually advertise routes to prefixes - stub ASes (the ones end users actually connect to) almost never advertise routes to IP addresses, unless they are the origin. Thus, we will never know to where stub ASes route without extrapolating - a technique which got us somewhere, but not far enough.

List of Figures

1 Histogram of Number of ASes tricked (by prefix) . . . 3
2 Histogram of Number of ASes tricked (by prefix, extrapolated) . . . 3
3 AS vs Number of Routes . . . 4
4 AS vs Number of Routes (one hop) . . . 4
5 Prefix vs. Num Tricked . . . 7
6 AS versus Number of Prefixes Fallen For (by size) . . . 8
7 Prefix vs Num ASes Tricked (extrapolated data) . . . 9
8 Prefix vs Num ASes Tricked (extrapolated data, zoomed) . . . 10
9 Histogram by Origin Size . . . 11
10 Histogram - AS vs Prefix by Origin Distance . . . 12
11 AS versus Number of Prefixes Fallen For (by distance from origin) . . . 13
12 AS versus Number of Prefixes Fallen For (by distance from attacker) . . . 14
13 AS versus number of Routes . . . 15
14 AS versus number of Routes Zoomed . . . 16
15 AS versus number of Routes (one hop) . . . 17

Appendix

In the following figures, AS sizes are split in one of two ways. If the graph refers to prefixes, then they are grouped by Origin AS size (unless otherwise specified).

1. ALL : All ASes are included
2. SMALL : ASes with 20 or fewer neighbors
3. MED : ASes with 21 - 100 neighbors
4. BIG : ASes with more than 100 neighbors (but not a T1 AS)
5. T1: Tier 1 ASes (6461, 7018, 2686, 5623, 3549, 3356, 701, 702, 703, 2914, 3561, 1239, 6453, 209, 3561, 3320 , 2828, 6762, 3257, 1299, 12956, 6461, 1299)

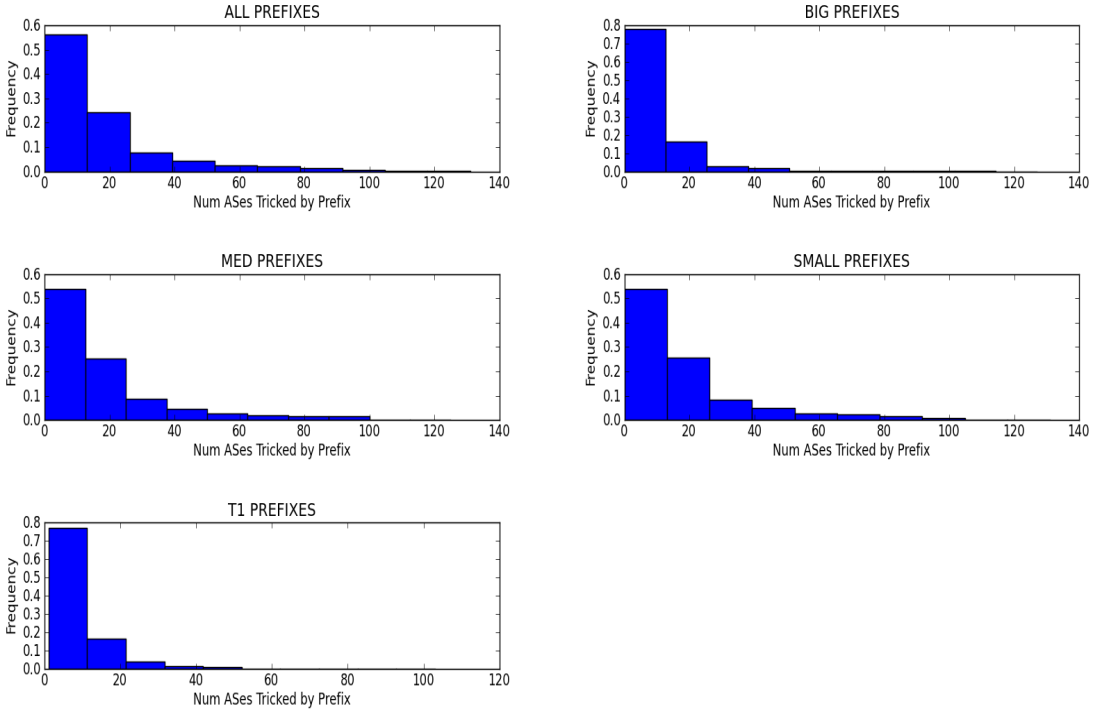
OR

1. <20 : ASes with 20 or fewer neighbors
2. 20+ : ASes with 21 - 50 neighbors
3. 50+ : ASes with 50 - 100 neighbors
4. 100+ : ASes with 100 or more neighbors (but not T1)
5. T1: Tier 1 ASes (6461, 7018, 2686, 5623, 3549, 3356, 701, 702, 703, 2914, 3561, 1239, 6453, 209, 3561, 3320 , 2828, 6762, 3257, 1299, 12956, 6461, 1299)

Prefixes versus Number of ASes Tricked (directly from data)

Figure 5

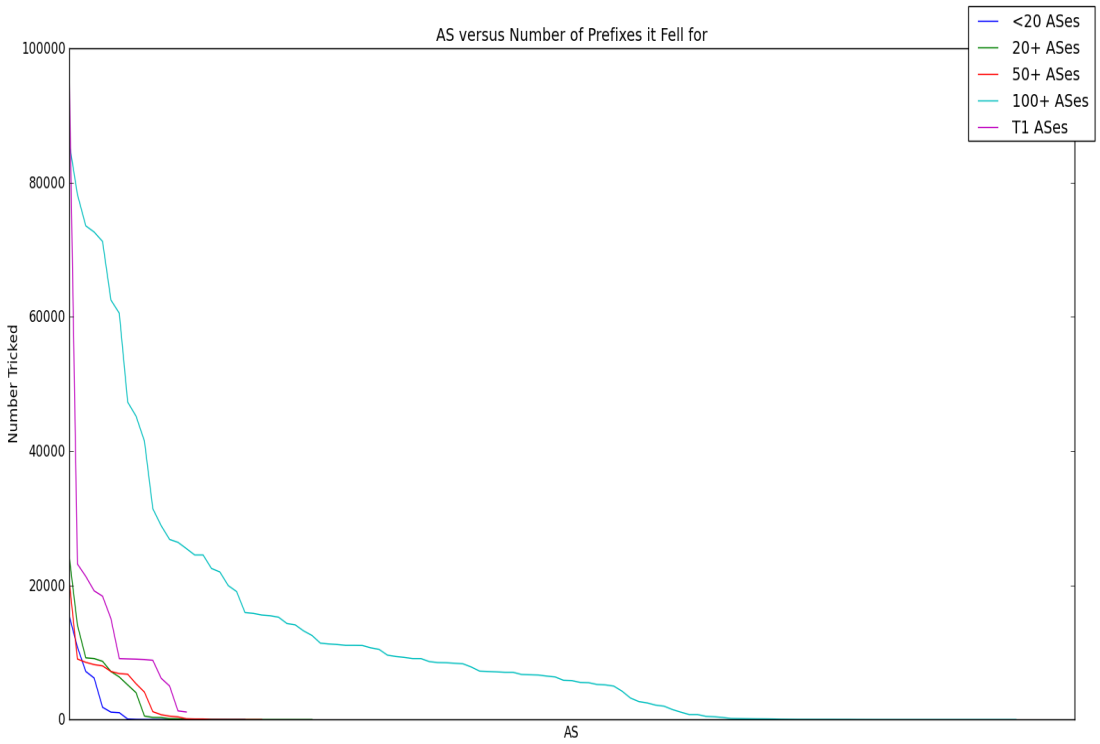
This section represents our first pass at analyzing the data - comparing each prefix versus the number of ASes which were tricked during the day. This is a naive comparison, using only the ASes that we observed on routes leading to the attacked AS.



Note the small number of ASes: in the dozens instead of the thousands as we expected to see. This lead us to attempt to extrapolate which other ASes might have been tricked.

Figure 6 - AS versus Number of Prefixes Fallen For (by size)

This plot shows AS versus total number of prefixes it was tricked on, the datasets were ordered from greatest to least for comparison.

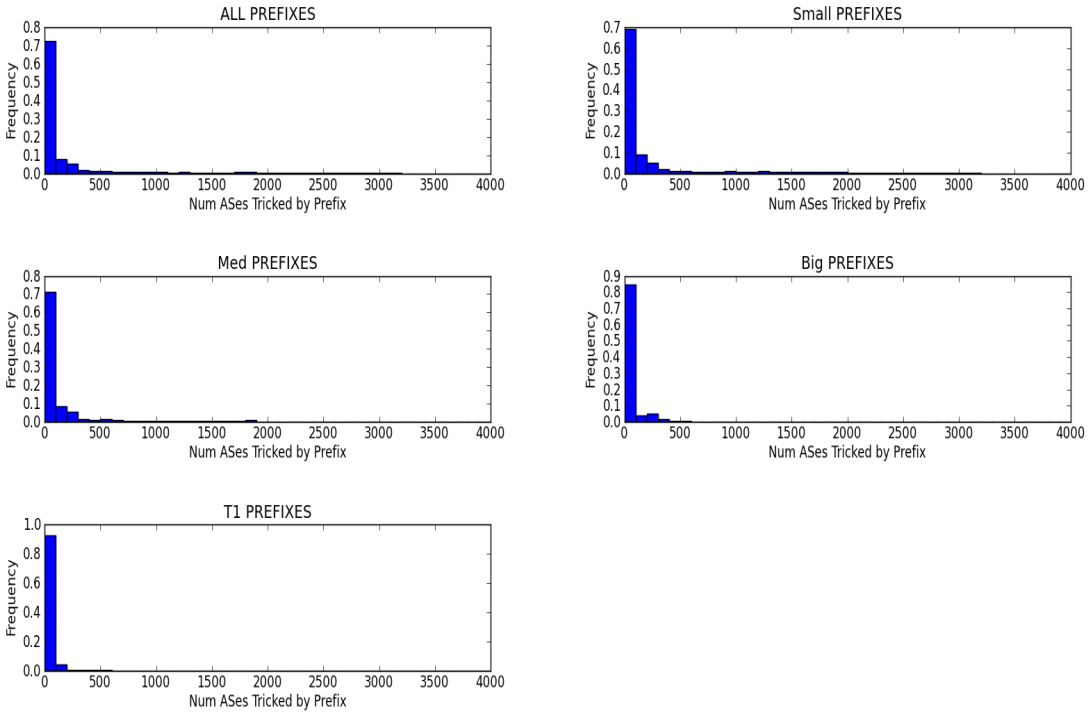


Note: the x-axis here is different for each line, every point on the line is a different AS. Interestingly, we see here that the largest ASes appear to be tricked on the most routes. However, this graph is not entirely convincing because of our vantage point. The biggest ASes send out the most updates, but a small stub AS may change its route without letting anyone else know. This line of thinking led us to try and extrapolate which other ASes may have been tricked.

Prefixes versus Number of ASes Tricked (with extrapolated data)

Figure 7

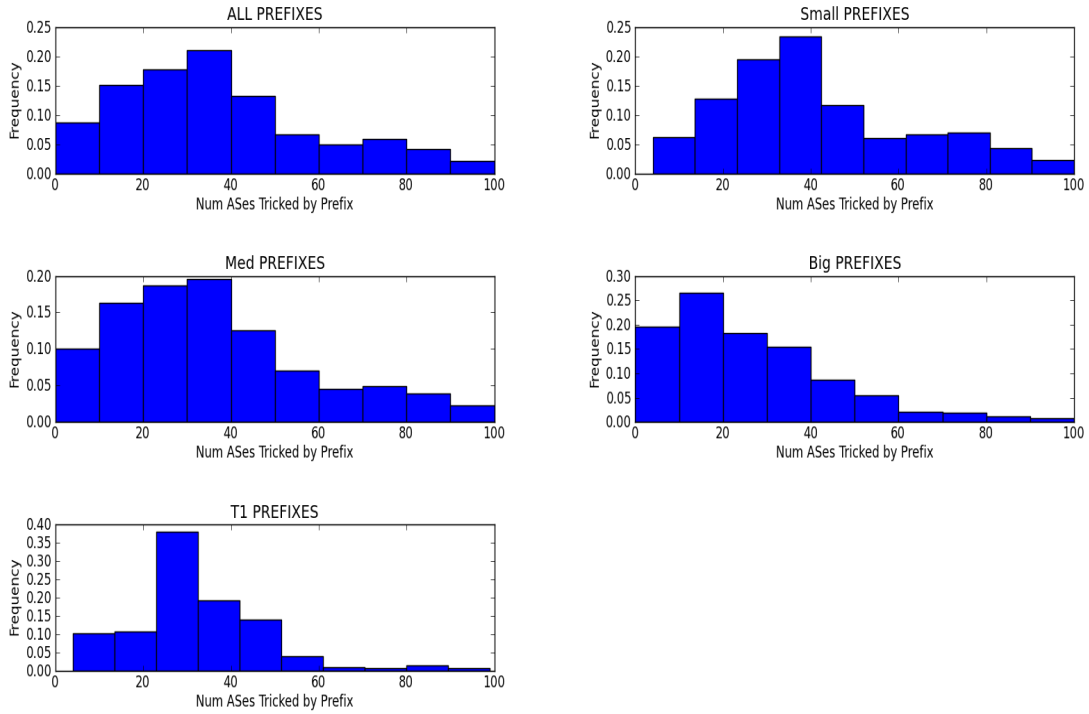
These histograms are generated in the same manner as above, with one exception. We inflated the number of attacked ASes by analyzing an AS graph from 2004. After generating the list of tricked ASes from the BGP update data, we looked at all ASes and added any AS to the 'tricked' set if all of its neighbors were also tricked. For example, any stub with a single provider is classified as tricked if its provider is tricked. We believe this is an extremely conservative metric, and that any numbers we see here are a lower bound.



Note: the number of ASes inflates significantly - even into the thousands.

Figure 8 - zoomed to under 100

Here we zoom in on Figure 7 to 100 or less ASes tricked.



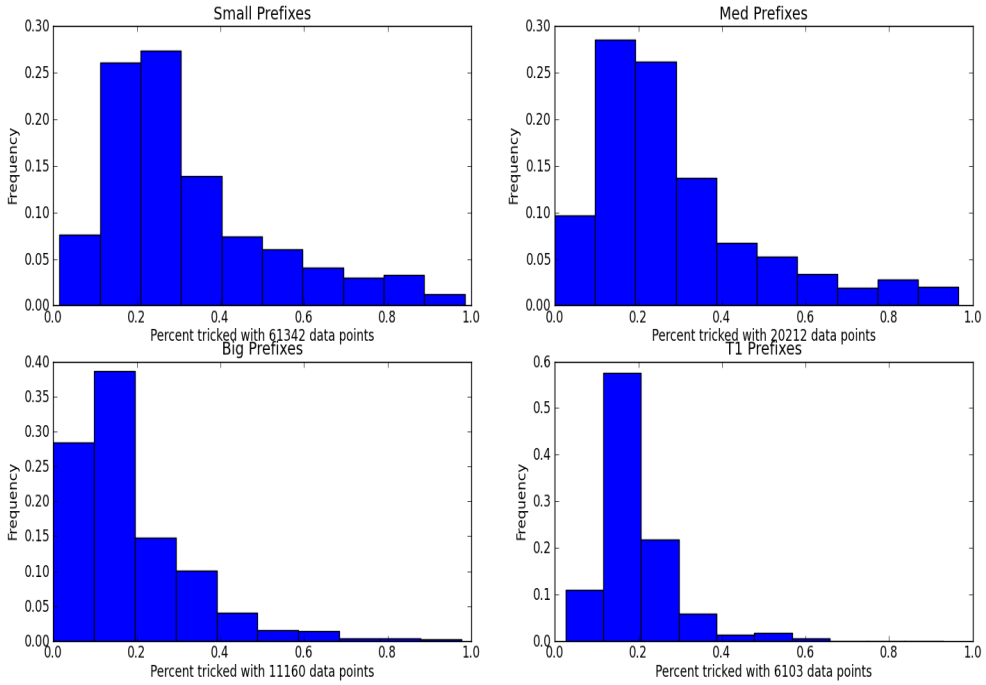
This graphic implies that most prefixes ended up tricking between 40 and 100 ASes. Based on the severity of the attack, and that "for a large number of Internet users, some chunks of the Internet were unreachable for at least a few hours on the morning of December 24 last year" [11], we believe that our data doesn't fully represent the extent of the attack. We believe the reason for this phenomenon is the way in which the data is collected. RIPE and Routeviews each host multiple "collectors" next to as many ASes as will let them. These collectors sit on one link between two ASes and records everything that goes back and forth on the link. Unfortunately, there are not nearly as many collectors (on the scale of dozens) as there are links in the internet (millions). While we can infer quite a bit about the routing going on in the internet from our vantage point, it is clear to us that there are many pieces which are missing.

Data Grouped by Origin Size

Figure 9 - Frequency

To generate these plots, we classify prefixes by the size of their origin.

This figure plots the distribution of the relative frequency of ASes which were tricked. For example, a prefix falls into the .2 bucket if 20% of the ASes we observed on routes relating to that prefix went to the attacker.

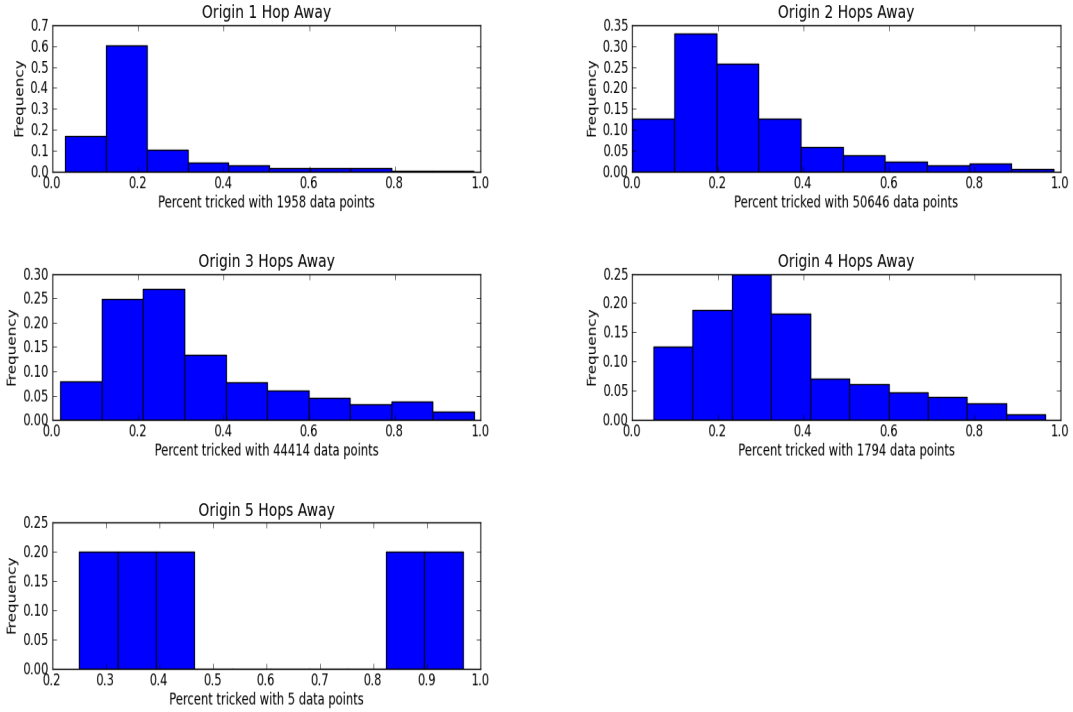


We draw two conclusions from these plots. First, it doesn't matter how big the origin AS is, the distribution looks roughly the same. Second, only a small fraction of ASes appear to fall for this attack.

Data Grouped By Distance

Figure 10 - Histogram - AS vs Prefix by Origin Distance

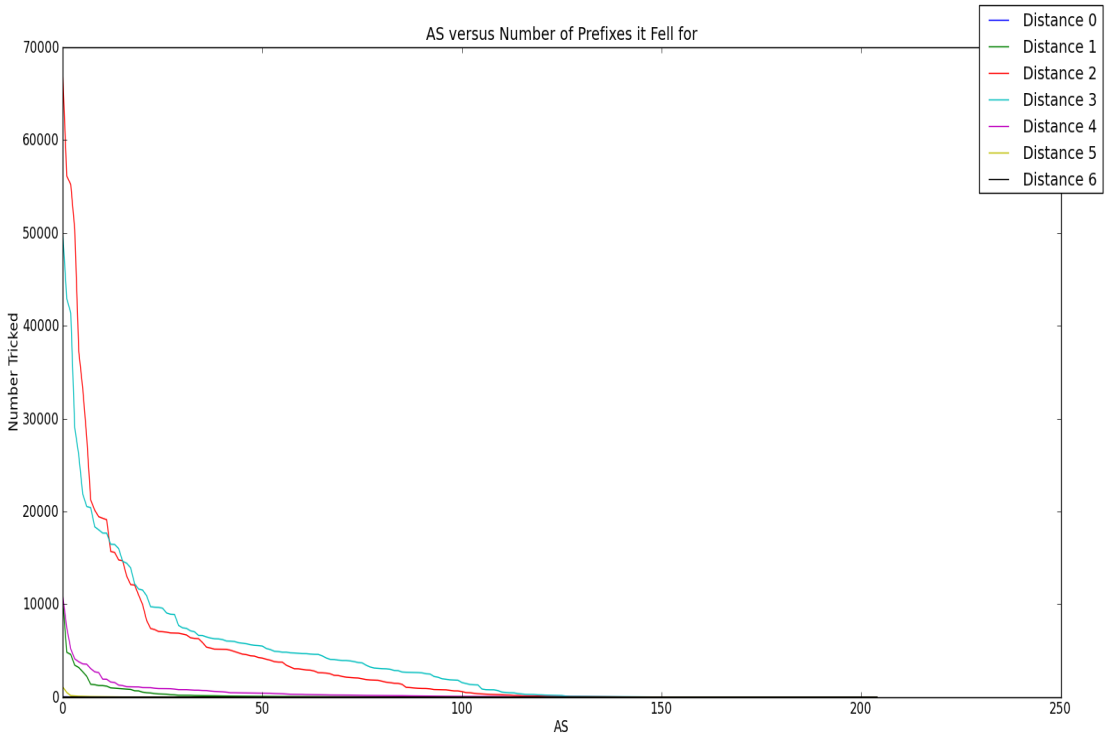
These plots show the relative frequency distribution again, but with prefixes grouped differently - instead of origin size we use attacker distance from origin.



The conclusions we draw here are the same as above, the distance doesn't appear to affect the number of ASes which are tricked.

Figure 11 - AS versus Number of Prefixes Fallen For (by distance from origin)

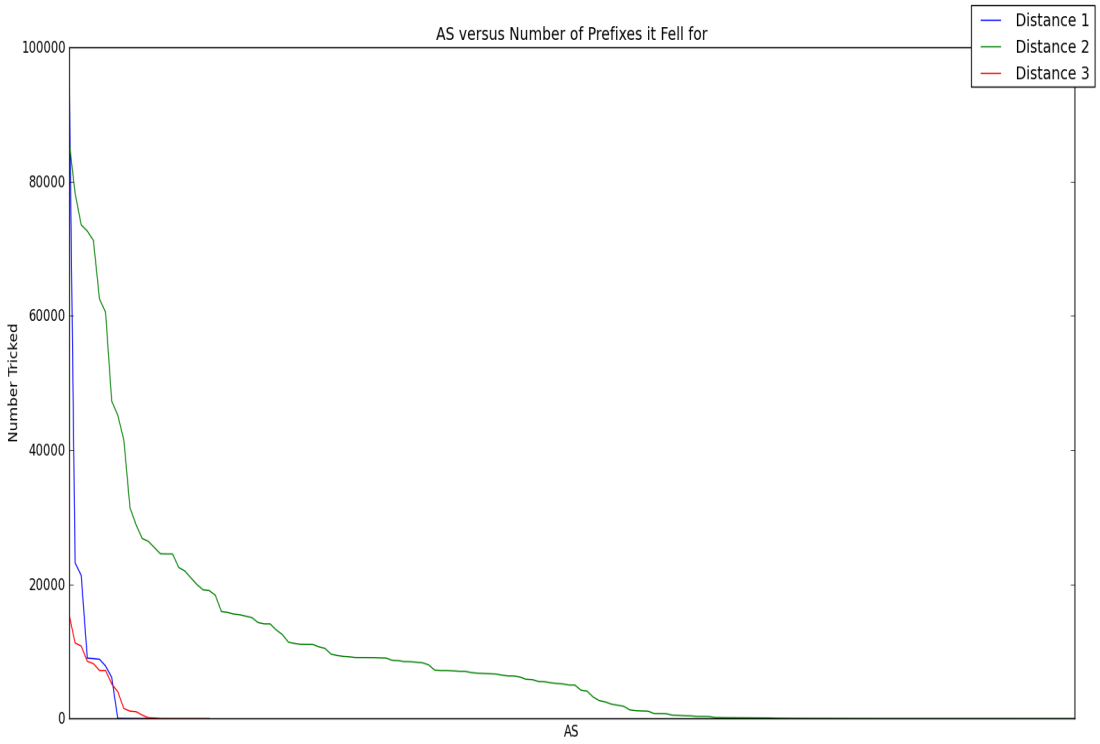
This plot shows AS versus total number of prefixes it was tricked on, by distance from the origin AS.



While conclusions are difficult to draw based only on this graph, there is a clear trend - its most likely that an AS will fall for an attack if its more than one hop away. This makes sense, ASes closer to the origin are least likely to change routes, as their existing routes are already short.

Figure 12 - AS versus Number of Prefixes Fallen For (by distance from attacker)

This plot shows AS versus total number of prefixes it was tricked on, by distance from the attacker AS. Note: the x-axis here is different for each line, every point on the line is a different AS.



This graph sends a clear signal, ASes which were two hops from the attacker were most likely to fall for the attack. We believe this is a meaningful result, although we are hesitant to make strong claims using only this as evidence.

Data Grouped By Number of Tricked ASes

Figure 13 - AS versus number of Routes

This figure represents AS versus number of mentions in tricked routes. For each AS we collected every route it was mentioned in which led to the attacker (a tricked route). Each data point represents the number of tricked routes a specific AS showed up on, aggregated by the size of the tricked AS set.For example, the blue line represents the total number of times an AS was mentioned for all IP addresses which had at least 1000 or more ASes tricked (normalized by the number of IP addresses in the set). Our goal was to try and quantify whether or not any specific AS was highly correlated with widespread adoption of a bad route.

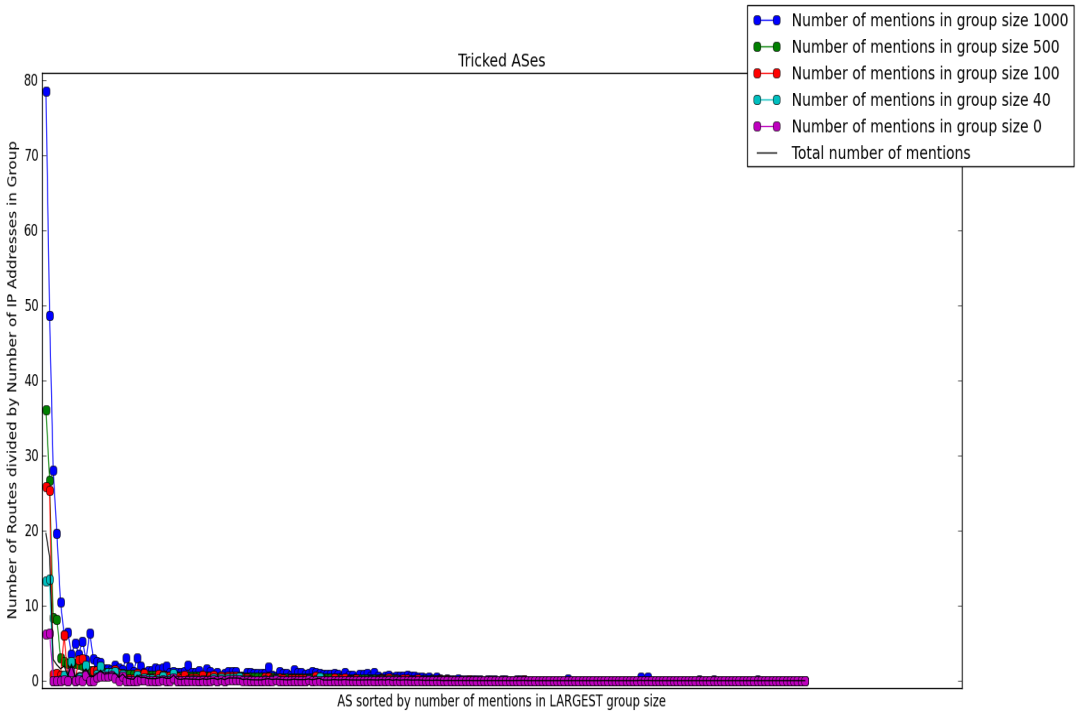
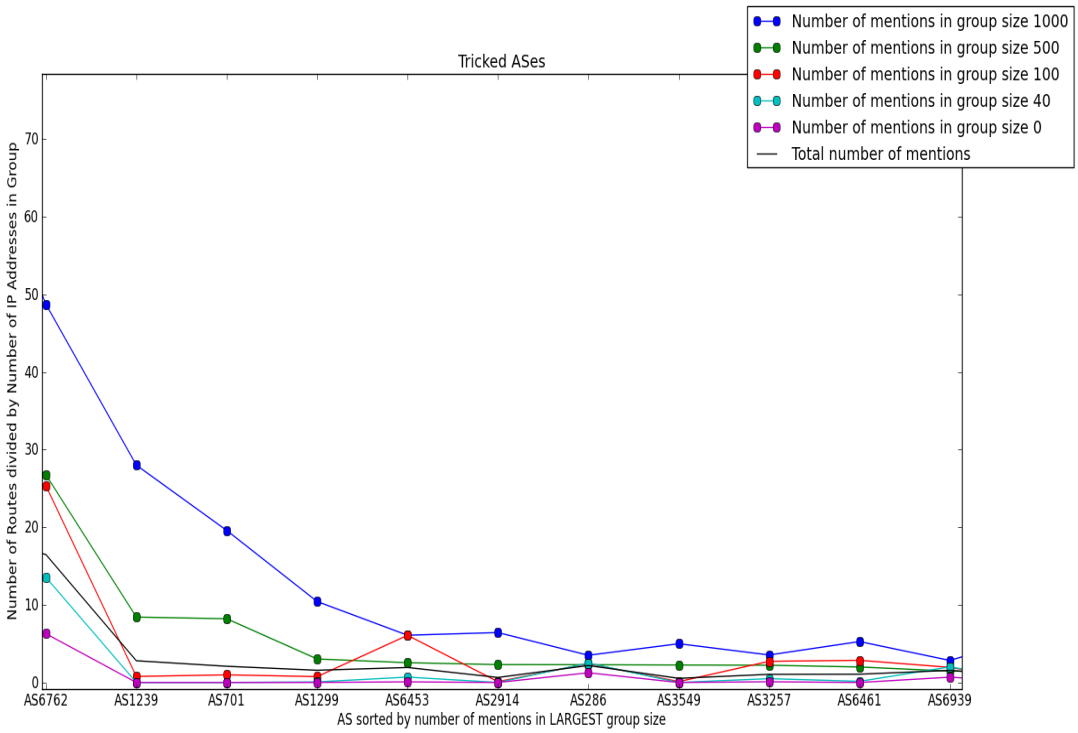


Figure 14 - AS versus number of Routes Zoomed

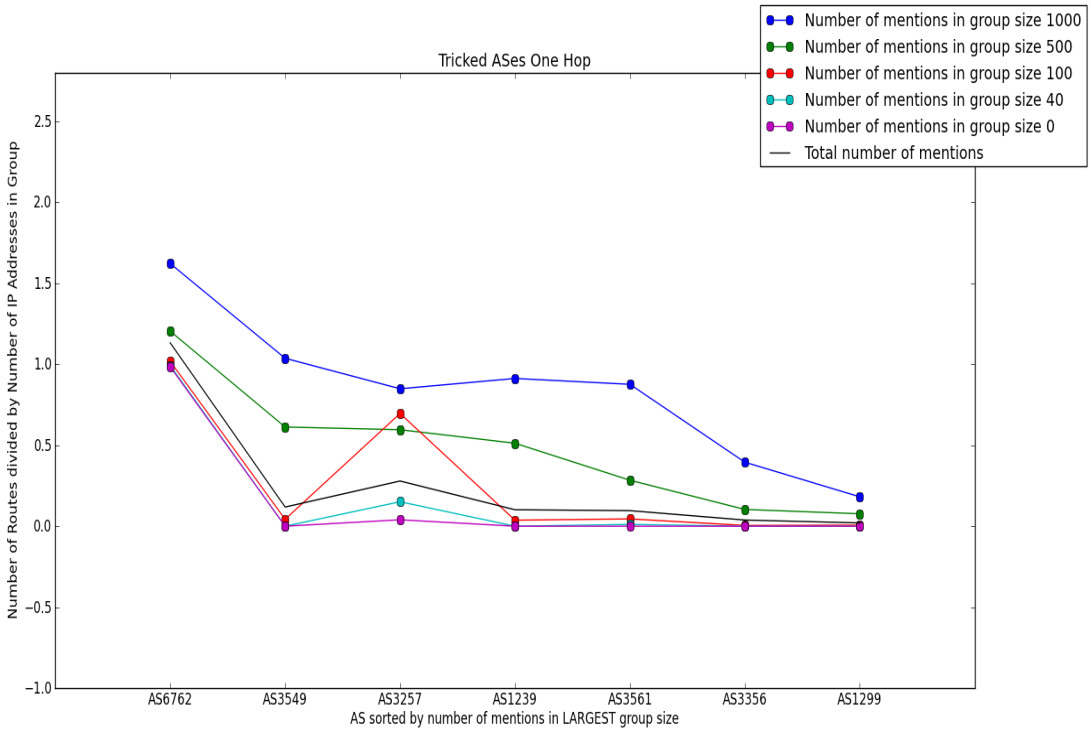
Here we see the same graph above, but zoomed in on the ASes which were most 'tricked'.



We can infer a relatively concrete observation from this graph: AS 6762 shows up in most routes - this is to say that if AS6762 was tricked, it is more likely that other ASes would also be tricked. However, one source of bias in this plot is that ASes closer to the attacking AS, AS9121, are more likely to show up on routes. This observation led us to graph the same data but only include ASes which are one hop from 9121.

Figure 15 - AS versus number of Routes (one hop)

This figure shows the same data from above, but only includes the 7 ASes which are exactly one hop from AS9121



Note that we observe roughly the same effect here, AS6762 is highest correlated with bad routes.

References

[1] Cyclops. http://cyclops.cs.ucla.edu/

[2] G. Huston. Peering and settlements - Part I. The Internet Protocol Journal (Cisco), 2(1), March 1999

[3] G. Huston. Peering and settlements - Part II. The Internet Protocol Journal (Cisco), 2(2), June 1999.

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

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

[6] RIS Raw Data. http://www.ripe.net/data-tools/stats/ris/ris-raw-data

[7] Routing Report. http://thyme.apnic.net/

[8] Routeviews. www.routeviews.org

[9] R. Lychev, S. Goldberg, M. Schapira. BGP Security in Partial Deployment. Is the Juice Worth the Squeeze? In Sigcomm, 2013

[10] S. Goldberg, M. Schapira, P. Hummon, and J. Rexford. How secure are secure interdomain routing protocols. In Sigcomm, 2010.

[11] T. Underwood. Internet Wide Catastrophe Last Year. Renesys Blog: http://www.renesys.com/2005/12/internetwide-nearcatastrophela/. Dec. 24, 2005