Detecting BGP Hijacks on the Internet

INTERNET PROTOCOOLS

MUHAMMAD GUMILANG, MERRILL SHEN, AND JESSICA YUAN

Github Repository

1. Introduction

BGP hijacking occurs when attackers maliciously reroute Border Gateway Protocol (BGP) traffic on the internet. An autonomous system (AS) can carry out a BGP hijack by falsely announcing ownership of groups of IP addresses that they do not actually own, control, or route to [1]. These groups of IP addresses are called IP prefixes.

The overarching goal of this project is to observe route announcements made by ASes on the Internet, develop tools that help detect when BGP hijacks occur, and examine the implications our results have on Internet routing at large.

The four types of hijacks examined specifically in this project are: multiple-origin AS (MoAS), sub-MoAS, fake path and DEFCON 16. MoAS hijacks occur when an AS announces itself as the origin of a prefix that it does not own, while a sub-MOAS hijack is when the AS announces itself as the origin of a sub-prefix it does not own. A fake path hijack occurs when the AS keeps the origin AS intact but announces a fake AS path to its neighbours. A DEFCON 16 hijack occurs when an AS announces a true path with a more specific prefix.

2. Methodology

2.1 Ground Truths

To detect hijacks, we need to know three ground truths:

- **True Owner of a Prefix:** MoAS hijackers announce a prefix as if it were their own. Thus, knowing the true owner of a prefix would allow us to identify the hijacker AS, which is necessary for MoAS detection.
- **True Prefix Size**: Sub-MoAs hijacker ASes announce, as their own, a subset of addresses of a prefix they do not own. Thus, knowing the true prefix size of an AS would allow us to identify when a sub-prefix is being announced, which is necessary for Sub-MoAs detection.
- **Real links between ASes**: Knowing which links between different ASes exist on the internet is necessary for fake path detection.

As for DEFCON 16 hijacks, knowing all three ground truths is essential for detection.

2.2 Training Dataset

BGPStream is an open-source software framework that provides access to streams of route updates from different route collectors [2]. We leveraged BGPStream to capture BGP announcements made by different ASes on the internet. We use this as training data to establish the true origins of prefixes and the AS paths with which they are typically seen. Learning this information is essential for detecting hijacks.

The training data used was from a 7-day period in the year 2017, and was read from two different BGP route collectors, one located in Singapore and one in Ashburn, Virginia [3].

2.3 Model Training

For each BGPStream update, we extracted the following information:

- peer_address: the IP address of the AS that announced the update
- peer_asn: the AS number of the AS that announced the update
- as-path: the AS-path from the peer to the origin
- prefix: the IP address of the prefix announced by the origin

By monitoring these fields and learning from the BGP routes commonly announced in our training dataset, we established heuristics for the three ground truths described above [4].

In the training data, we discovered that multiple ASes announced the same origin of a prefix. Since a unique prefix can only belong to one AS [5], we assigned the true owner of a prefix to be the AS that announced the prefix the most number times in the training data.

We assigned the true prefix size of a prefix to be the smallest prefix size observed in the training data. A sub-MoAS hijack increases the netmask number of a prefix to hijack a subset of the prefix. Thus, the lowest netmask number announced for a prefix should not come from a sub-MoAS hijacker. We decided to omit keeping track of different netmask numbers announced by each ASN for each prefix as we opined that the increase in accuracy was not worth the trade-off of a longer run-time and increased memory usage.

We implemented the described approach for finding the true owner and true prefix size by maintaining a prefix dictionary where keys are prefix addresses and values are dictionaries themselves. For each prefix, the second-level dictionary has 3 keys: ‘counter’, ‘mask’, and ‘asn’.

The ‘counter’ key maps to a Counter object that keeps count of all the AS networks (ASn) associated with the prefix and the number of times the ASn and prefix were announced together. For each BGPStream update, we update the prefix dictionary with the peer_adr and its corresponding peer_asn. We also update the prefix dictionary with the prefix and the corresponding origin ASn, which is determined from AS-path. Both updates increment the Counter object within the ‘counter’ key.

The ‘mask’ key maps to the lowest subnet mask observed in the training data for that prefix. Lastly, the ‘asn’ key maps to the ASn of the true owner of the prefix. The ASn that announced the prefix with maximum count in the ‘counter’ key is the ASn of the true owner.

We assigned true links between different AS networks to be any link observed in the AS-paths of the training data, which relied on the assumption that all AS paths observed in the training data set were valid.

We implemented true links by building a graph, where nodes are ASes and edges are links between ASes that were extracted from all the AS-paths in our training data. Thereafter, each link of an AS-path could be cross-referenced with the graph to verify its validity.

Once the dictionary and graph had finished training, implementation of the four types of BGP hijacks was straightforward. For MoAs and Sub-MoAs, the dictionary was used to lookup the true origin of prefix and true prefix size. For fake path, the graph was used to check for invalid edges between ASes. And for DEFCON 16, both graph and dictionary were used to check for valid edges between ASes and prefix size.

2.4 Testing Hijack Detectors

To put the detector implementations to the test, we ran all hijack detectors on the 30-minute BGPStream that followed immediately after the time frame used for the training dataset.

Each announcement of the BGPStream was fed to each detector to determine whether any hijacks occurred or not. Withdrawal announcements and announcements that contained prefixes that did not appear in the trained prefix dictionary were ignored. A detector that detected a hijack logged the time taken, the ASN that hijacked, and the hijacked ASN.

3. Results

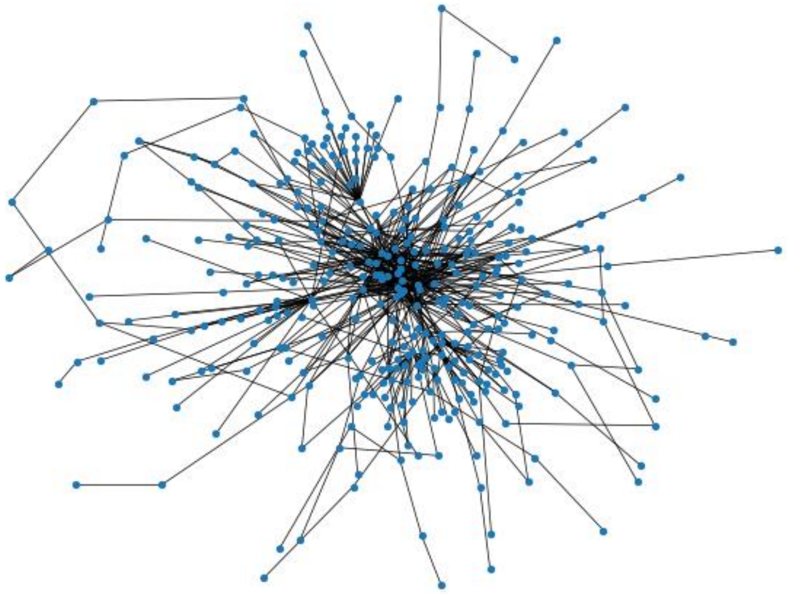

Fig. 1. Graph of ASes (Nodes) and Its Paths (Edges) From a 10-Minute Training Stream

Figure 1 illustrates the graph obtained from training with 10 minutes of BGPStream data where each node represents an AS and each edge represents the true link between two AS networks. We could not produce the graph for seven days of training data as our systems often timed out before the graph could be fully drawn. Overall, the graph consists of 59,103 AS networks and 132,001 links.

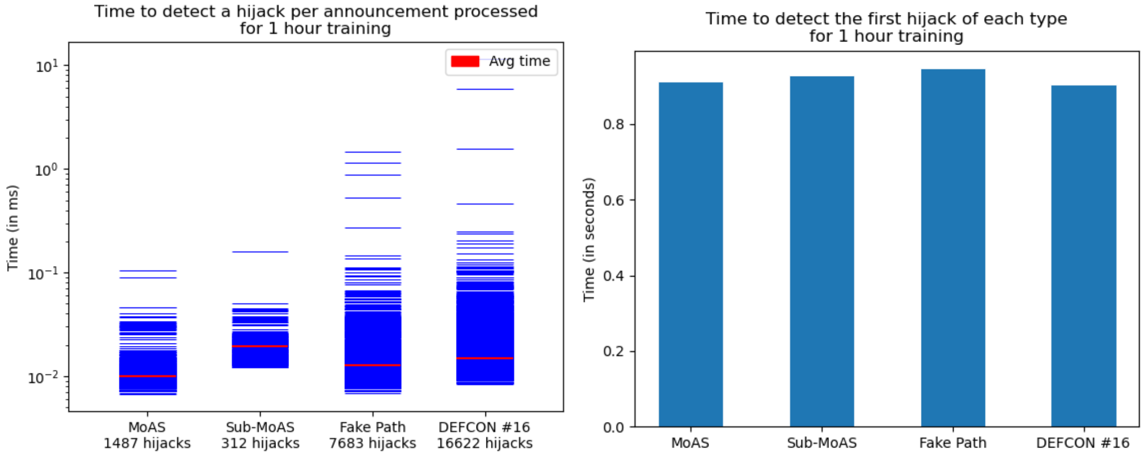

Fig. 2. Time Graph of Hijack Detections Using a 1-Hour Training Model

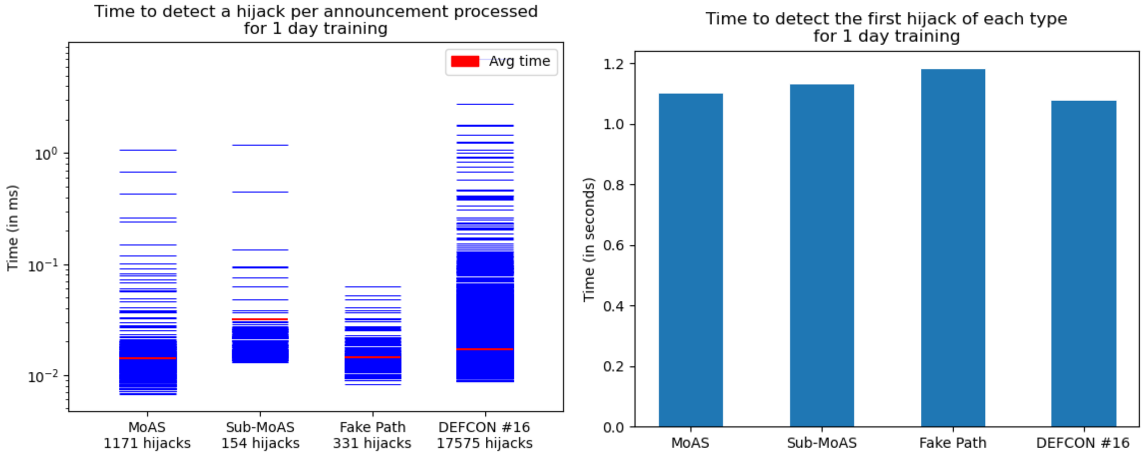

Fig. 3. Time Graph of Hijack Detections Using a 1-Day Training Model

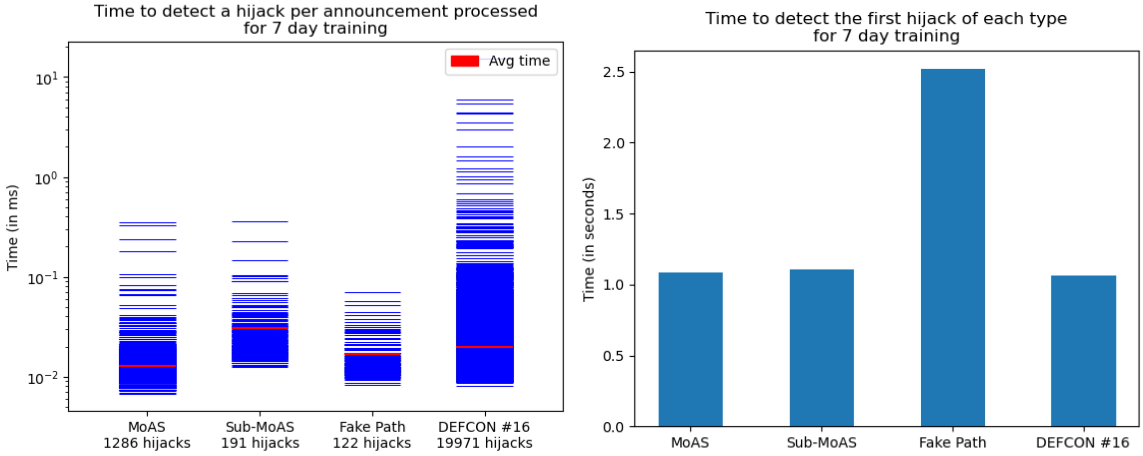

Fig. 4. Time Graph of Hijack Detections Using a 7-Day Training Model

The figures above show the average time taken to detect the first of each type of hijack, as well as the average detection time taken per announcement based on varying amount of training data. Figure 2 represents the metrics recorded with 1 hour of training data while Figure 3 shows 1 day of training data, and Figure 4 shows 7 days of training data.

A trend observed was that despite increasing the amount of training data, the time taken to detect the first hijack of each type stays relatively consistent except for Fake Path hijacks which increased. This could be attributed to that fact that a larger training data set would expose the model to more possible links between AS networks. Together with the assumption that all links in the training data set are valid, this produces a more complete graph which may take more time to fully traverse to verify if a link from the test set is truly valid.

As for the detection time for each hijack per announcement, the increasing amount of training data resulted in fewer false positives for fake path detection. The number of fake path hijacks decreased from 7683 to 122 hijacks when the training data set increased from 1 hour to 7 days of training data. The number of MoAS and sub-MoAS hijacks stayed relatively consistent, while the number of DEFCON 16 hijacks was found to increase with more training data. This could be due to our methodology of determining the true prefix size. Having taken the smallest prefix size announced in the training data set for each AS network as the true prefix size and not accounting for potential changes in the true prefix size, this would result in a greater number of DEFCON 16 hijacks.

Hijacking AS	**Hijacked AS**	**Hijacking Frequencies**
AMX Argentina S.A., AR (19037)	Techtel LMDS Comunicaciones Interactivas S.A., AR (11664)	274
DNIC-ASBLK-01513-01518, US (1516)	DNIC-ASBLK-01550-01601, US (1600)	240
No longer exists (16971)	VIPNAS1, VI (14434)	118
AIRTEL-, RW (327707)	CELTEL-DRC, CD (37020)	90
REACHONE, US (14517)	MASHELL-TELECOM, US (20394)	90

Table 1. Five Most Frequent Pairs of Hijacking AS And the AS it Hijacked.

Looking deeper into the actors of hijacks, the top pair of hijacking and hijacked ASes for MoAS hijacks were gathered from testing data and summarised in Table 1 above. Interestingly, these pairs were found to be within the same domain, except for the third pair where the hijacking AS no longer exists. The most frequent pair (274 times) are both telecom providers and so are the fourth and fifth pairs. Though we are uncertain of the intent behind these hijacks, we hypothesize that these telecom providers may announce the same prefix as they operate in the same geographical area, and both provide coverage to the same telephone number prefix. Another hypothesis may be because customers can switch their telecom provider but keep the same phone number. Hence, the prefix may be announced by the old provider as the new provider may take some time before everything is transferred over.

4. Limitations

The decision to assign the true owner of a prefix to the AS network that announced to be the origin of a prefix the greatest number of times may not be a reliable choice. However, to know the actual mappings between AS and prefix would require insight into proprietary intellectual property owned by AS networks which we do not have access to.

Implementing DEFCON 16 correctly would require access to information on previous announcements received by certain AS. This means a large amount of memory space would need to be allocated. Thereafter, we can decide whether an AS actually increased the netmask number by comparing with the past announcements they got.

5. References

[1] Cho, Shinyoung, et al. "BGP Hijacking Classification." *2019 Network Traffic Measurement and Analysis Conference (TMA)*, 2019, https://doi.org/10.23919/tma.2019.8784511.

[2] King, Alistair. *BGPStream*, bgpstream.caida.org/. Accessed 29 Apr. 2023.

[3] *Autonomous System numbers*. American Registry for Internet Numbers. (n.d.). https://www.arin.net/resources/guide/asn/

[4] Orsini, C., King, A., Giordano, D., Giotsas, V., & Dainotti, A. (2016). BGPStream: A Software Framework for Live and Historical BGP Data Analysis. *Proceedings of the 2016 Internet Measurement Conference*. https://doi.org/10.1145/2987443.2987482

[5] *Routing Information Service (RIS)*. RIPE Network Coordination Centre. https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris