Measuring the propagation of BGP Hijacks

Albert Williams, Arjun Sreedharan

College of Information and Computer Sciences, University of Massachusetts Amherst

Introduction

Problem statement

- The Internet's architecture relies on trust based peering arrangements between autonomous systems. This honor system that uses Border Gateway Protocol (BGP) could be exploited by malicious elements who announce IP addresses that do not belong to them..
- We identify IP addresses announced by more than one autonomous system (AS) and establish a heuristic to decide which are hijacks.
- We also establish which autonomous systems are propagating paths to which conflicting origin, letting us decide which AS can reliably ignore hijackers and which should not be trusted to tell the difference. This can help neighboring ASes choose between competing destinations, mitigating the effect of a BGP hijack.

BGP prefix announcement example

Resources

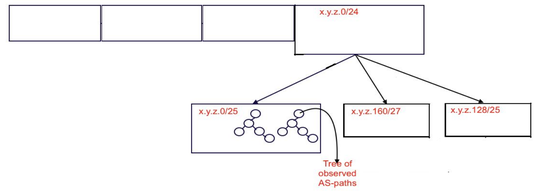
- BGP announcement data is collected from the BGPStream project (bgpstream.caida.org) run by the Center for Applied Internet Data Analysis (CAIDA).
- The *libBGPStream* and *PyBGPStream* APIs are used to access the stream of BGP data within a given time period.
- The BGPStream collector used for the study is RIPE RRC 10.

Challenges

- Finding ways to correctly identify if BGP overlaps are hijacks, or due to traffic engineering, or due to misconfiguration or possible filtering or censoring.
- In case of hijacks, finding good heuristics to distinguish the real owner of the IP from the hijacker using the details of conflicting announcements.

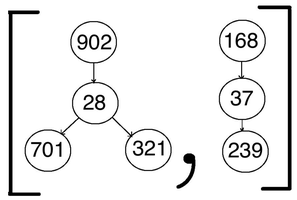
Methods

Infrastructure



The PrefixForest Data structure

- In order to easily analyze BGP Announcements, we record the data in a special data structure named *PrefixForest*.
- It has at the first level a list of *PrefixNode* objects.
  - *PrefixNode*s contain one *Prefix* object which corresponds to the least specific (shortest mask) IP address prefix that does not conflict with prefixes of other *PrefixNode* objects in the list.
- The children of a *PrefixNode* are *Prefix* objects that conflict with the it but have more specific IP prefixes.
- Upon seeing a new announcement for an existing prefix, the information from that announcement is merged into the existing *Prefix* object.
- Each *Prefix* object contains:
  - The value of the address of the IP prefix (e.g. 178.20.16.0/21).
  - The length the IP prefix.
  - A list of *ASPathTree* objects

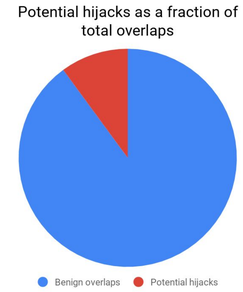


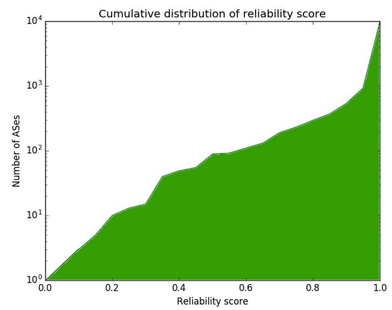
This is a typical list of ASPathTree objects corresponding to 3 overlapping BGP announcements received for a prefix and containing the following AS-paths:
701 28 902
321 28 902
239 37 168

- Once *ASPathTree* list is constructed, we are able to tell whose announcement each AS trusted for the same prefix.
- In the above example, the same prefix was announced originally by both AS 902 and AS 168. AS 37 chose to trust AS 168 while AS 28 chose to trust AS 902.

Results

- BGP hijacks were detected using the following heuristic:
  - With a *PrefixForest* data structure, groups of overlapping BGP announcements are detected.
  - All the ASes from which overlapping BGP announcements originate for a particular prefix are easily collected.
  - We classify any AS that announces a majority of the overlapping prefixes as the legitimate origin, and any others as hijackers
  - We classify no ASes as hijackers if there is no AS advertising the majority of prefixes





- BGP data between Aug 25 and Sep 24, 2017 were analyzed
- 0.48 % of all AS announcements were found to be potential hijacks
- Of all the overlaps detected, 10.07 % were found to be potential hijacks

- We devised a metric named **reliability score** to score downstream ASes based on if they propagated the announcement of the hijacker or that of the AS owning the prefix.
- If an AS has a reliability score of 1, it means the AS has never propagated a hijacker's announcement. This score reflects the trustworthiness of an AS from the perspective of downstream ASes.
- The global AS graph can be colored using this reliability score, and this can be used as a metric to help avoid BGP hijacks

Future Work