https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

MODELLING RELATIONSHIPS IN AUTONOMOUS SYSTEMS AND INTER-DOMAIN ROUTING POLICIES

Hardik Joshi[1], Arnav Gupta[2]

[1]Sr.Software Engineer, Proficient Business Systems
[2]Sr.Software Engineer, Cognizant Technology Solutions

Abstract

In this two part project, we plan to understand the relation-ships between Autonomous Systems (ASes) on the Internet and derive inferences between theoretical and practical rout-ing scenarios. For the rst part of the project, we use the March 2016 CAIDA Dataset [3] that represents a theoret-ical model of the AS relationships using the now standard Gao-Rexford model [1] and create a graph visualization us-ing Neo4j [4] graph database. We present our analyses using statistics obtained with this dataset and pick the top 100 mostly connected ASes and use them for further analysis. In the second part of the project that we will progress next, we will explore practical routing scenarios using RIPE Atlas framework by using probe nodes found in the rst part and nd out routing instabilities at inter-AS routing level. We concern ourselves with the problem of AS Prefix Hijacking and present a simulation of the same using MiniNet.

***Keywords:** - Autonomous Systems; Gao-Rexford Model[1]; RIPE Atlas; BGP; Graph Database*

---***---

1. PROBLEM STATEMENT

Internet Measurement - modeling relationships in Autonomous Systems (ASes) and Inter-domain routing policies using CAIDA dataset and RIPE Atlas.

2. INTRODUCTION

2.1 BGP and Autonomous Systems

In the Internet, different ISPs communicate with each-other. These ISPs communicate with their customers us-ing various Intra-domain routing protocols whose decision is made by the provider itself. They can decide the protocol based on their network trace and distribution of customers in the geographical area (topology). However all these ISPs have to communicate with each other via a common protocol. This protocol which is agreed by the entire Internet community is called Border Gateway Protocol (BGP). The interdomain (exterior) routing paradigm is driven by the BGP communication between the Autonomous Systems (ASes). These Autonomous Systems are the exit point of a particular network driven by a network provider. These ASes communicate with other ASes based on certain policies. These policies are varied and are explained by now stan-dard and accepted Gao - Rexford model.

2.2 Gao - Rexford Model

Generally while running a protocol to setup routes to the destination; it is typically done in a dynamic fashion via path vector-based implementation. However this protocol complexity lies in the set of routing policies each A Simplements. To converge to a particular destination, there are often different ASes to traverse. A standard approach may be selecting the shortest path or the least cost path. However this is not true in all the cases, in practice the selection of ASes is based on various other factors including but not limited to nancial policies between ISPs and amount of tra c to traverse the network. Due to this, there are al-ways anomalies in the shortest path and the selected path in practical scenarios. Gao - Rexford model provides a set of guidelines that an AS can implement to set up its routing policies without coordination with other ASes and guarantee the convergence of path. This model gives us the ideal path between the ASes. We then compare the actual path and nd anomalies in BGP routing in this project. We ob-tain the ideal connection details between ASes from CAIDA data set and the Actual connection using RIPE Atlas.

2.3 CAIDA Dataset

CAIDA stands for Center for Applied Internet Data Analysis. CAIDA curates datasets resulting from both active and passive Internet measurement and has been used in several research papers. The dataset establishes relationships using the Gao-Rexford model. We use this information provided by CAIDA and obtain a list of ASes that are critical to the Internet infrastructure. Later we work to obtain the AS Topology map using this dataset.

2.4 RIPE Atlas

RIPE Atlas is a global, open, distributed Internet measurement platform, consisting of thousands of measurement devices that measure Internet connectivity in real time. Each AS has a probe sitting at the node to measure the network

2.4.1 Probes

Probe is a small hardware device that runs measurements in the RIPE Atlas system and reports the measurement results

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

to the data collection components. This probe is hosted by a host for RIPE Atlas; that is, someone who applies for a probe connects it to their network and leaves it running. Hosting a RIPE Atlas probe benefits the entire measurement community. Hosting the probe earns credits for the time that the probe is connected. These credits can then be used to conduct user-defined measurements using the entire RIPE. These probes collects measurements like Current up-time, total uptime and uptime history, RTT, Ping measurements, Trace route measurements etc.

2.4.2 Anchors

RIPE Atlas anchors are both enhanced RIPE Atlas probes with more measurement capacity, as well as regional measurement targets within the greater RIPE Atlas network. Anchors provide valuable information about the local and regional connectivity and reachability of the Internet, and the large amount of data they collect is made available to everyone.

2.4.3 RIPE Atlas Streaming API

RIPE Atlas offers a number of ways to conduct Internet Measurement. There is a public set of measurements that constantly monitor traffic and can be used for analysis. We can also have a User-Defined Measurement (UDM) using a probe or an anchor. For example using a UDM, we can Ping another server on the internet or find it's Traceroute or DNS servers, etc. However all these functionalities re-quire RIPE credits which can be obtained by payment or by hosting your personal machines as RIPE probes and gaining credits over time. Another approach to analysis is using the archived dumps provided by RIPE, but these dumps need parser libraries which are not updated regularly. We tried using existing set of libraries to parsing the dumps but were unable to make them work with newer datasets. More recently RIPE started hosting live-streams of the measurements using a RESTful API interface. Using socket.io protocol, it is pretty easy to establish a connection in JavaScript whereby we start receiving the measurements for a particular probe and measurement ID. We can replay historical events using a start date and an end-date and also retrieve metadata about the status of the probes.

2.4.4 BGP Prefix Hijacking

BGP Prefix hijacking is the illegitimate takeover of groups of IP addresses by corrupting Internet routing tables. Two hosts in the Internet communicates with each other with IP address. This is achieved by passing data from one router to another until it is safely delivered. To do this, each router must be regularly supplied with up-to-date routing tables. At the global level, individual IP addresses are grouped together into prefixes. These prefixes will be originated, or owned, by an autonomous system(AS) and the routing tables between ASes are maintained using the Border Gateway Protocol (BGP). Each AS uses BGP to advertise prefixes that it can deliver traffic to. Prefix hijacking can occur deliberately or by accident in one of several ways:

1. An AS announces that it originates a prefix that it does not actually originate.
2. An AS announces a more specific prefix than what may be announced by the true originating AS.
3. An AS announces that it can route trace to the hi-jacked AS through a shorter route than is already available, regardless of whether or not the route actually exists.

Various methodologies to overcome this BGP prefix hijacking has been devised. Some of the key insights about these techniques includes improved real time detection systems to detect malicious attacks, implementing route filtering at eachnode etc. Some advanced versions of BGP like SBGP have also been designed by researchers to improve the performance of BGP and eliminate the prefix attack threats.

2.5 Visualization Tools

2.5.1 Neo4j

Neo4j is one of the existing graph databases that can be easily hosted on personal machines. Typically huge graph visualizations are difficult to handle on a browser due to main memory constraints but Neo4j is an exception as it limits the display settings according to user's requirements and underlying hardware. Other visualization tools such as Tableau and Gephi are independent desktop applications. For our experiments, we imported the set of ASes as nodes and two sets of AS relationships viz. Peer-Peer and Customer-Provider from the CAIDA dataset into Neo4j.

2.5.2 D3.JS

D3.js (or just D3 for Data-Driven Documents) is a JavaScript library for producing dynamic, interactive data visualizations in web browsers. It makes use of the widely implemented SVG, HTML5, and CSS standards. It is the successor to the earlier Protovis framework. In contrast to many other libraries, D3.js allows great control over the final visual result. It allows you to bind arbitrary data to a Document Object Model (DOM), and then apply data-driven trans-formations to the document. Using D3.js, we could create a time-varying graph as routes changed depending on the traceroute hops.

2.5.3 Ip-api.com

We needed a mechanism to retrieve the ASN given an IP on the Internet. Ip-api.com is one such way to solve this problem. It is a GEO location API which is used to give in-formation regarding the geographical location, country, ASN number and other metadata. Using AJAX in JavaScript, we could asynchronously call this API each time a new node is input in the D3 graph.

2.5.4 Mininet

Mininet is one of a kind software emulator which works on simulating a large network on a single machine. The unique characteristics of Mininet are that it can be used to create a

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

realistic virtual network running actual kernel, switch and software application code on our own computer. It helps user to quickly create, interact with, customize and share a software-defined network (SDN) prototype and hence simulate a network topology that uses Open-Flow switches.

Mininet provides an easy way to understand the appropriate system behavior and to experiment with different topologies. In mininet we have run real code including standard Linux network applications as well as the real Linux kernel and network stack. This is the reason that the code we have developed and tested on Mininet, for an Open Flow controller, modified switch, or host, can be easily moved to a real system with minimal changes, for real-world testing, performance evaluation, and deployment. The major take-away here is that a design that works in Mininet can usually move directly to hardware switches for line-rate packet for-warding.

Mininet creates a new network namespace for each host in the simulated network. Also Mininet starts the switch and controller processes in the Mininet VM's root which are just processes running on the Mininet VM. It is also possible to set up the controllers and the switches each in their own network namespace so they operate as separate virtual machines networked to each other across virtual Ethernet interfaces.

The main responsibility of mininet is that each of VM will be able to communicate with any VM on the same switch. Thus it provides commands using which we can measure and control each network element. Mininet forwards commands to the nodes in the network from the Mininet command line. The output of the command will be displayed on the Mininet terminal. Internally, Mininet employs lightweight virtualization features in the Linux kernel, including process groups, CPU bandwidth isolation, and network names-paces, and combines them with link schedulers and virtual Ethernet links. These features yield a system that starts faster and scales to more hosts than emulators which use full virtual machines. To perform more interactive tasks on the virtual host computer, such as editing configuration files or working with programs that have their own command-line interfaces, like Quagga. For our experiments, we used MiniNet to demonstrate AS Prefix Hijacking.

3. PROBLEM DESCRIPTION

From the early days of Internet, it has been constantly evolving. Now we see that millions of entities have been added over the period of time. The common man could only wonder how the existing entities connect with each other let alone the new ones. The protocol which has been assigned this responsibility is BGP. Although the routing of a packet as it happens it became increasingly clear to the entities that the current shortest-path routing is not sufficient to be able to manage the ever changing scenario which is constantly influenced by the operational, economic, and political factors involved in routing. The main driving force behind this modified decision making were the ISPs which began to change the routing configurations to manage the routing policies, which mean that the decision making done by the owners of the router which were managing the routes were changed and hence the routes which were exchanged between the neighbors were also different now. It was in the earlier days of BGP it followed a simple routing path-vector protocol. It is just subsequently that so many changes have been made by the ISPs so that they are able to reduce the cost they bear for routing packets to and fro through their own net-work. The protocol which now exists for routing packets is completely changed and lot of modifications introduced that are not compatible with each other and one can observe several conflict in various ways. The main problem with these updates is that many of the changes can be highly un-predictable and many of them are involved in the decision process which is important for selection of routes. Some of the modifications are not even mentioned in the standard protocol specification. It has become increasingly difficult to keep track or make any predictions of any sort as to how the packet will route and which path it will take. Thus the complexity increasingly causes increase in problems which might include security problems or wrong configuration and also the several conflicts while interacting between router owned by different ISPs which could break the Internet infrastructure.

The most important responsibility of autonomous systems is to send and receive the information about network routing between different ASes. The information being exchanged also composes of information about the ASes which were being traversed in order to reach the intended destination (hop information). We require this information in order to develop the path which was actually taken by the packet in order to reach the destination. We can derive lot of other information about the reachability using which certain loops can be discarded and also the ISP specific policy decisions could be implemented and studied. It is very important to understand the basic fundamentals and how the model of network routing form the base for implementing security and improving the reliability and constantly evolve as currently the experiments done to understand the routing are done using simulations of the entire routing system. It is very difficult that any model can be proposed based on some simulation as it will not be based on real time data, as we already know that it is the ISPs are constantly trying to take benefit and reduce cost by tweaking routing policies for their router and many don't even let the other companies know what actual routing policies they might have used for their routers.

Discrepancies or anomalies are a day-to-day occurrence for the inter-network routing protocol. They could include any-thing from a misconfiguration in the infrastructure to an in-

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

correct update being shared between the routers that could dangerously affect routing for particular regions. As a researcher thus it becomes a responsibility to understand and bring forth some of the routing anomalies that exist which can help us in understanding things like filtering, misdirection and interception. It entirely depends on the anomaly and which of the control/data plane. This detection can be done using a very sophisticated approach. It becomes very important to get the dataset from different ASes which form the key role in the routing of packets and which have more and more connections to other routers. It is the AS which has most number of probes is very important to study and understanding the routing the policies of such AS would help us find the anomalies that could help us the different policies employed by the ISPs. In order to find anomalies we need to be able to select the top ASes and to identify the anomalies at this ASes.

In this project, we have thought of analyzing AS routing scenarios by looking at theoretical standard model i.e. Gao Rexford model versus practical routing scenarios obtained by traceroutes via ATLAS RIPE APIs. The motivation for this is to ensure authenticity and consistency on the Internet. Initially we present our analysis on the well-known and heavily used CAIDA AS dataset and then proceed to its visualization using graph database. We try to derive inferences on the nature of the AS graph existing on the Internet using this dataset and identify possible ASes and probes that we can use for the next part of the project.

4. SOLUTION METHODOLOGY

We decided to split our project into two parts: AS Structure Visualization and AS BGP practical routing. For part 1, we used the CAIDA dataset to obtain AS relationships existing as of March 2016. We obtained two files - AS Rel file that contains AS-AS relationship mapping which can either be Peer to Peer or Provider to Customer. The other file contained a list of existing connections between ASes from where we identified most prominent ASes that has maximum connections. We retrieved a list of 100 most connected lists of ASes and found their corresponding relationships in the first file. This gave us a list of 28846 unique ASes that we visualized along with their existing Peer-Peer and Provider-Customer relationships using Neo4j.

5. EVALUATION AND RESULTS

5.1 Analysis on CAIDA Dataset

We first performed analysis on entire CAIDA dataset as of March 2016. The summary of this set is listed below:

Table 1: Statistics of Complete Dataset

No of Unique ASes	53537
No of Links	406401

We next narrowed this dataset by using the top 100 most connected ASes. We list summary and top 5 most connected Ases these below:

Table 2: Statistics of Reduced Dataset

No of Unique ASes	28846
No of Links	68712

Table 3: Top 5 most connected ASes

ASN	Name	Name
ASN3356	Level 3 Communications, Inc.	USA
ASN174	Cogent Communications	Cogent Communications
ASN1299	TeliaSonera AB	TeliaSonera AB
ASN2914	NTT America, Inc.	NTT America, Inc.
ASN3257	Tinet Spa	Tinet Spa

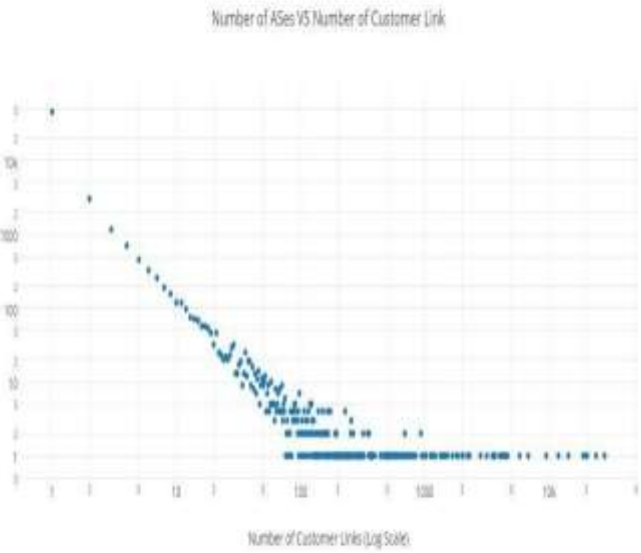


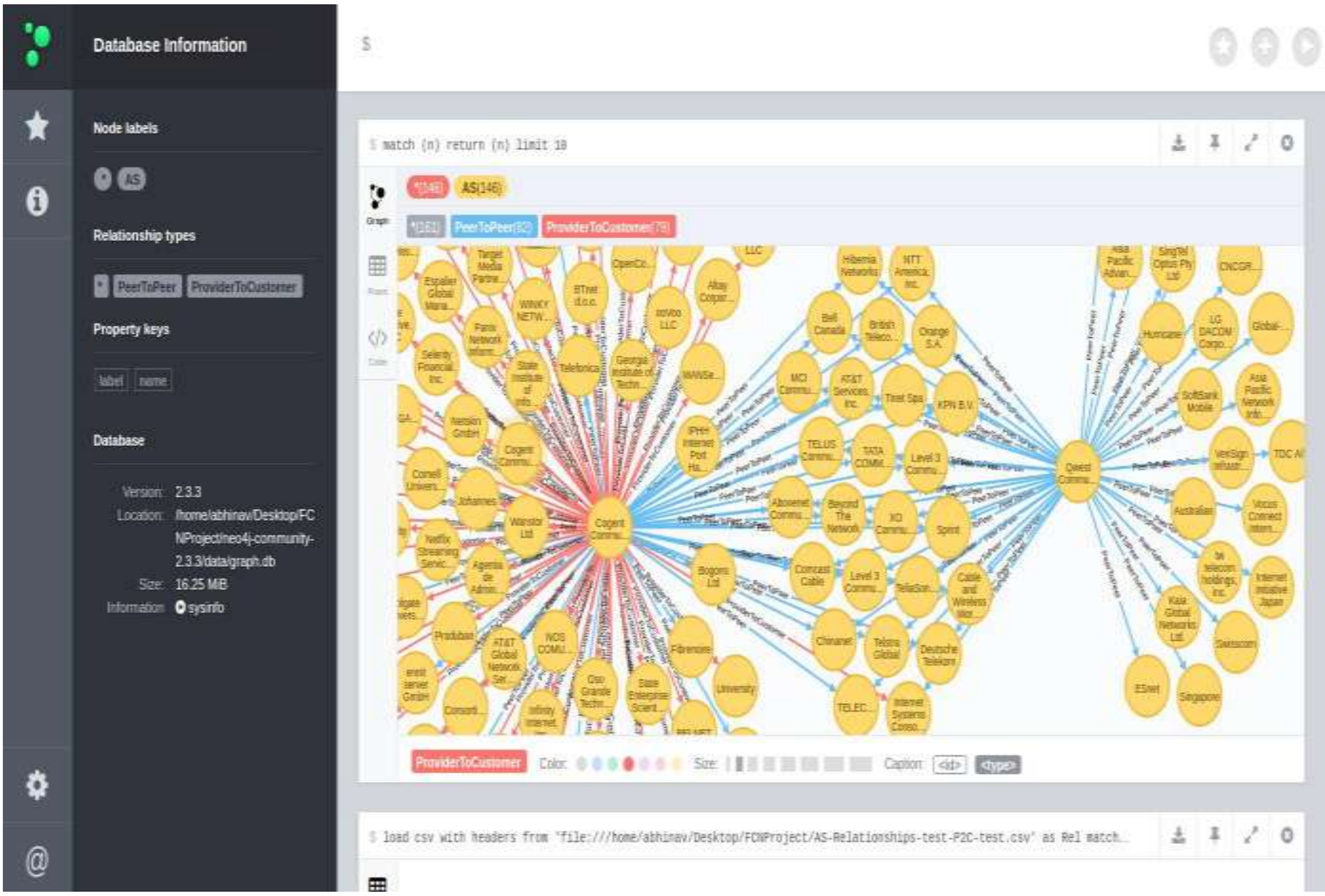
Fig 1: Scatter Plot of AS vs Number of links

The scatter plot above indicates that there are very few nodes having huge number of links (Peer-Peer and Provider-Customer). Around 15000 ASes have only one link while the highest number of links were around 28000.

5.2 AS Relationship Visualization and Inferences

Now we used Neo4j which is a graph database for visualizing these ASes. We created two CSVs containing the two relationships i.e. Provider to Customer and Peer to Peer. We initially created the total list of 28846 nodes in the database and then input the relationships files.

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018



Database Information

Node labels

Relationship types

Property keys

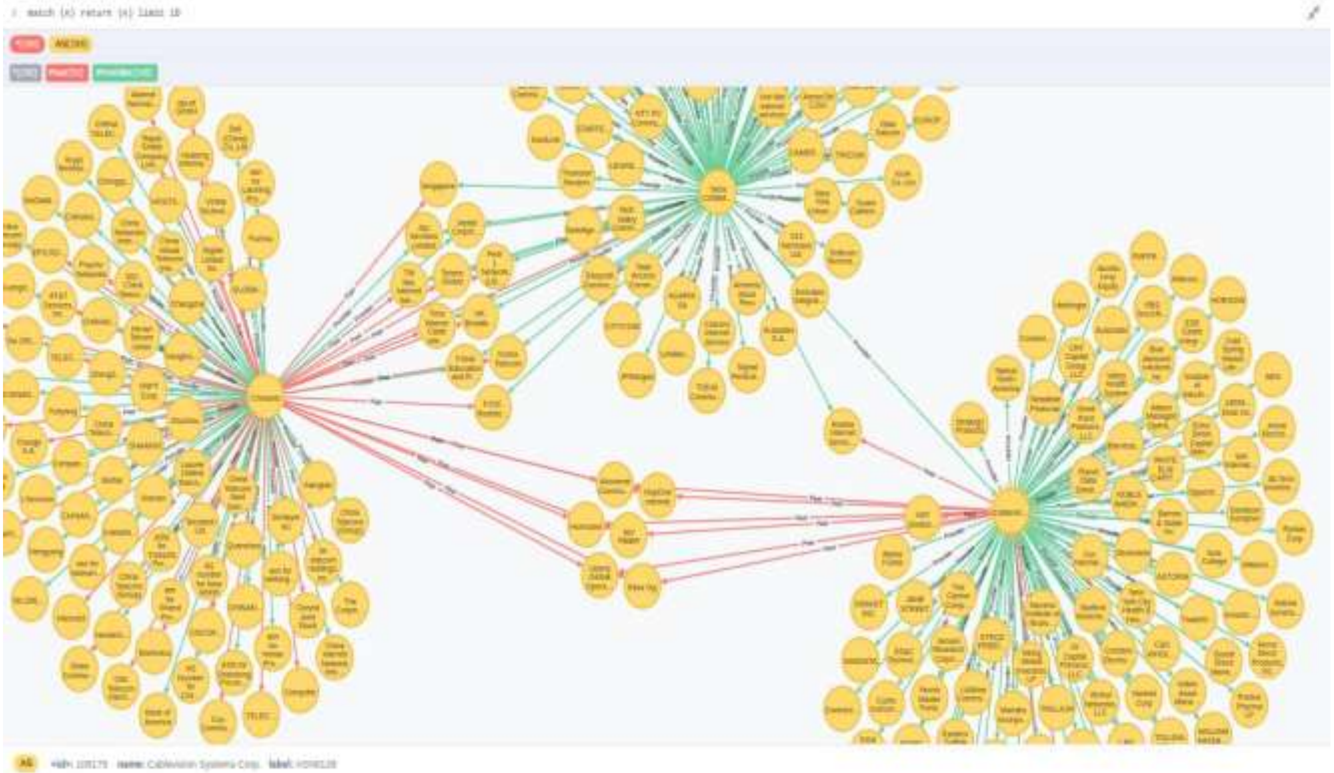
Database

Version: 2.3.3
Location: /home/abhinav/Desktop/FCNProject/neo4j-community-2.3.3/data/graph.db
Size: 16.25 MiB
Information: sysinfo

```
$ match (n) return (n) limit 18
```

```
$ load csv with headers from 'file:///home/abhinav/Desktop/FCNProject/AS-Relationships-test-P2C-test.csv' as Rel match…
```

Fig 2: Visualization of Cogent and Qwest Communication ASes



```
$ match (n) return (n) limit 18
```

Fig 3: Visualization of ChinaNet vs Cablevision vs Tata Comm (US)

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

One of the interesting results that we could see is that the more highly linked ASes have very few Peer-Peer relationships with other ASes compared to a larger portion of Provider-Customer relationships.

5.3 RIPE Atlas

Amongst the top 100 highly connected list of ASes, we identified three particularly interesting ones that can be helpful in our further analysis with RIPE Atlas - AS6128: Cablevision Systems, AS6453: Tata Comm (US), AS4134:

ChinaNet. For these three ASes, we found existing Internet Measurement probes as listed in the summary table below. We will use these measurement ids as our Vantage Points for our next experiments.

Table 4: Vantage Points

ASN	Name	Atlas Probe IDs
AS6453	Tata Comm (US) INC	19797
AS6128	Cablevision Systems Corp.	23478,22723, ..
AS4134	ChinaNet	23846,15504, ..
AS12637	Seeweb S.R.L(Italy)	726

5.4 Comparison between CAIDA and RIPE

We can now do a comparison between the practical and theoretical routing as we have enough information from both CAIDA and RIPE. Below table shows details of the CAIDA real time stream that we use in our experiments.

Table 5: Experimental Setup Information

ProbeID	726
Measurement Id	1663314
Source	ASN12637: Seeweb S.R.L.
Destination	AS39759: Passepartout S.P.A.

Using the traceroutes that we received, it can be seen that the practical route requires 4 hops i.e. 3 hops which are Customer-Provider and 1 hop which is Peer-Peer as shown in Fig. 4. While the CAIDA dataset gives us a much shorter path i.e. 1 Peer-Peer and 1 Customer-Provider as shown in Fig 5.Also the Customer-Provider link is preferred over Peer - Peer link for routing. This concludes that practical routing is indeed a acted by local policies in the AS and the theoretical models like Gao-Rexford cannot completely justify such routing scenarios. The images below show these relationships.

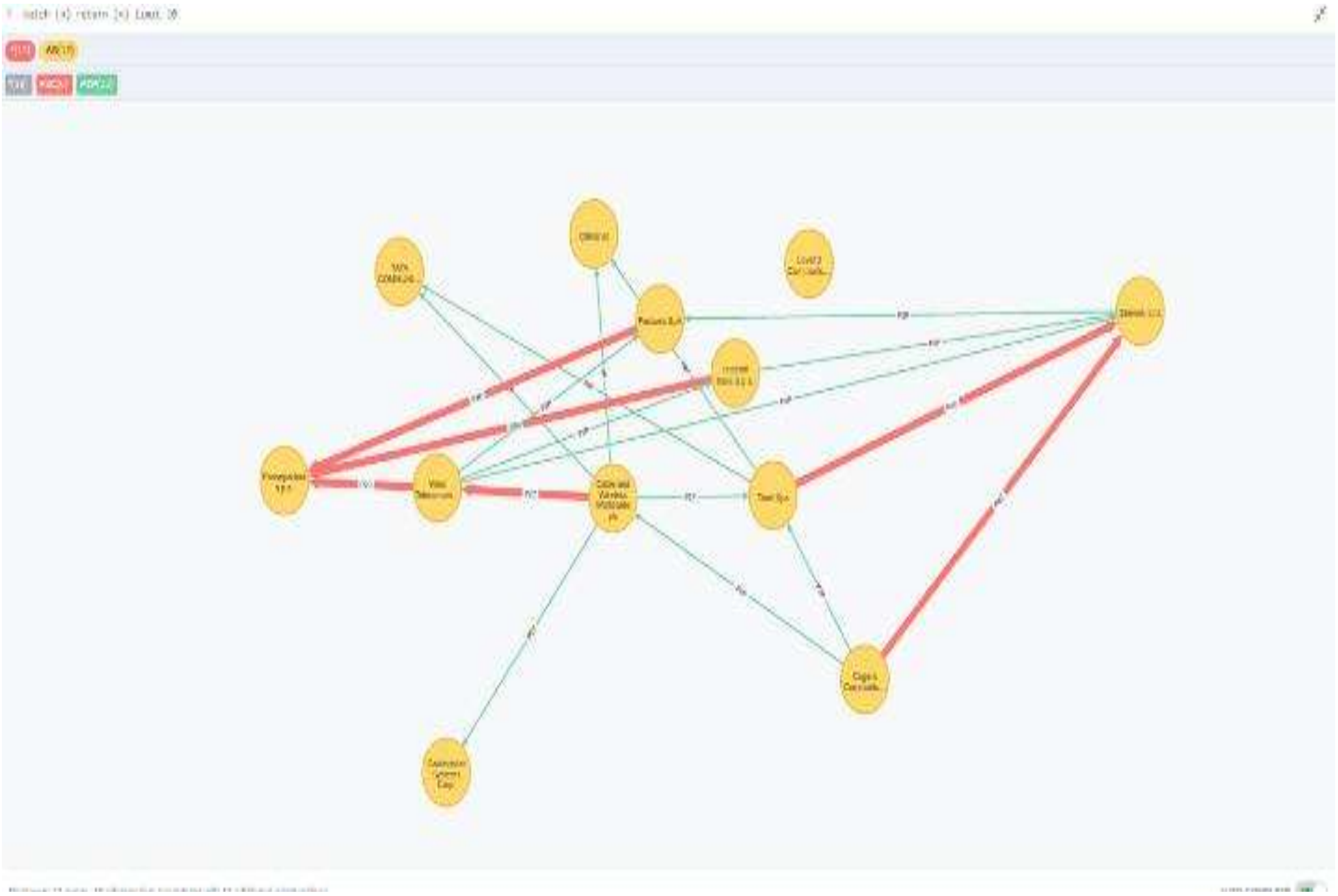


Fig 4: CAIDA Neo4j Experiment

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

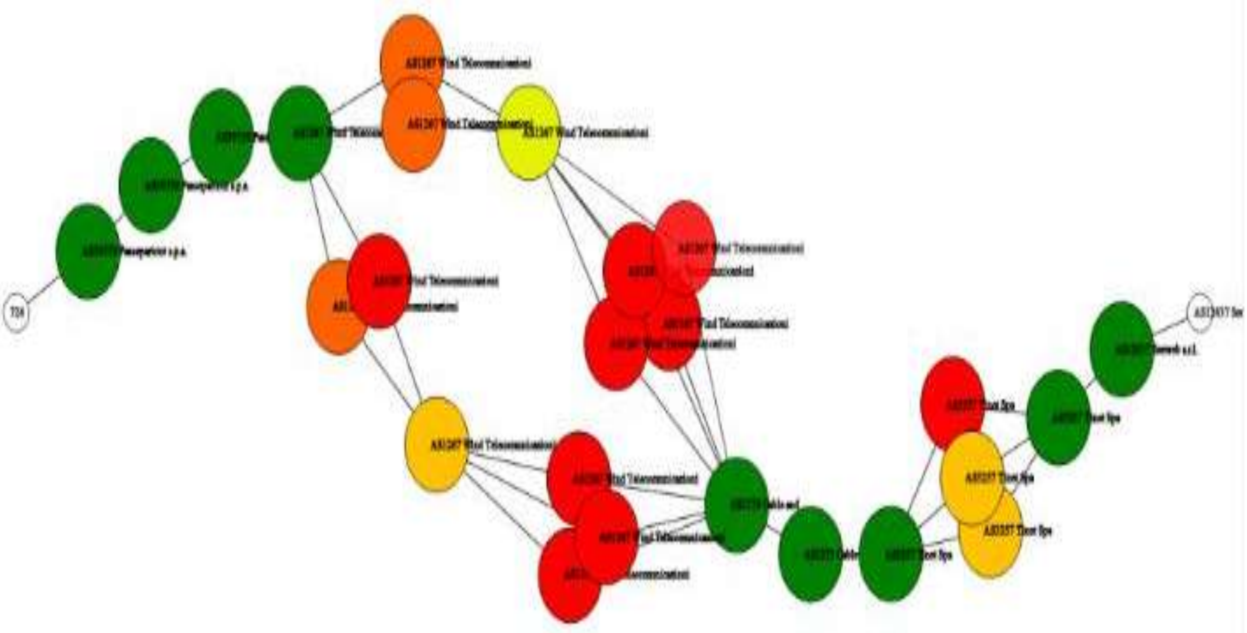


Fig 5: RIPE Atlas Streaming Experiment

5.5 BGP Prefix Hijacking

Using Mininet we created 4 Ases each with 4 hosts and 1 edge router. Each router Ethernet port was configured according to below mentioned topology and IP routing table. We considered AS1 as source and AS3 as the destination and

it also had AS4 as the attacker AS who had the same pre xes as AS3. Initially the communication took place between the source and destination until the Attacker AS started the rouge Pre x advertisement and asked the source to send the information to Attacker since the destination IP can be reached over a shorter route. Thus the packets were spoofed.

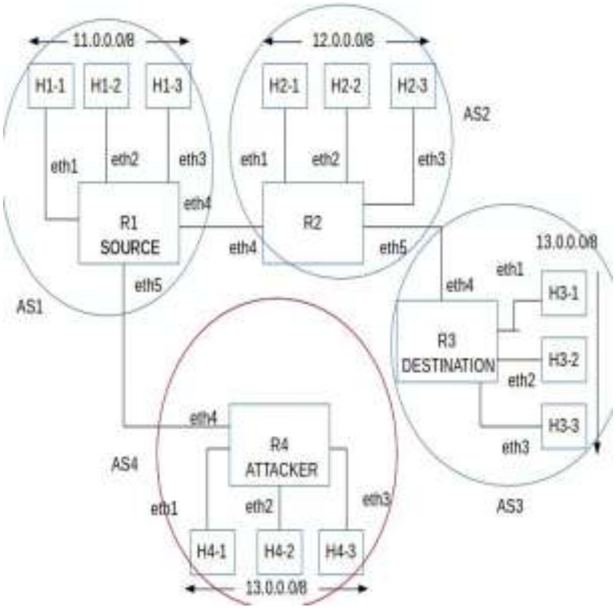
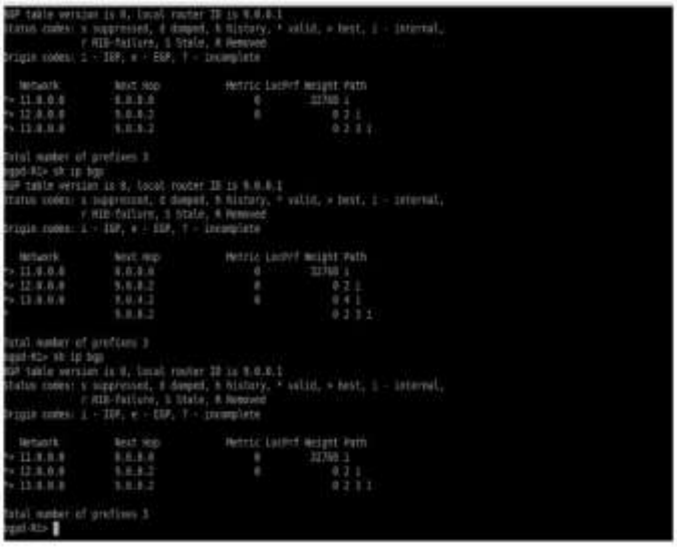


Fig 6: Prefix Hijack Mininet Topology

Later when the Attacker stopped the rouge advertisements the previous connection between Source and Destination was restored and the communication was intrusion free.



```
BGP table version is 0, local router ID is 9.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 11.0.0.0         0.0.0.0                  0         32768 i
*> 12.0.0.0         9.0.0.2                  0             0 2 i
*> 13.0.0.0         9.0.0.2                                0 2 3 i

Total number of prefixes 3
bgpd-R1> sh ip bgp
BGP table version is 0, local router ID is 9.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 11.0.0.0         0.0.0.0                  0         32768 i
*> 12.0.0.0         9.0.0.2                  0             0 2 i
*> 13.0.0.0         9.0.4.2                  0             0 4 i
*                   9.0.0.2                                0 2 3 i

Total number of prefixes 3
bgpd-R1> sh ip bgp
BGP table version is 0, local router ID is 9.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 11.0.0.0         0.0.0.0                  0         32768 i
*> 12.0.0.0         9.0.0.2                  0             0 2 i
*> 13.0.0.0         9.0.0.2                                0 2 3 i

Total number of prefixes 3
bgpd-R1> 
```

Fig 7: IP Con guration

6. CONCLUSION

Through this project we have explored theoretical routing models de ned at the inter-AS level using Gao Rex-ford model and analyzed different AS relationships using well known CAIDA dataset. We then investigated practical routing scenarios using real time streaming API provided by RIPE Atlas and found useful inferences between the two. We also explored instabilities related to BGP that can cause issues viz. the AS prefix hijacking using a simulation of routing tables in MiniNet.

https://doi.org/10.15623/ijret.2018.0701003 Received: 17-11-2017, Accepted: 04-01-2018, Published: 26-01-2018

REFERENCES

[1] Stable Internet Routing without Global co-ordination" - Lixin Gao, Jennifer Rexford
[2] Investigating Interdomain Routing Policies in the Wild" - Ruwaifa Anwar, Haseeb Niaz
[3] CAIDA datasets - http://data.caida.org/datasets
[4] Neo4j Graph Database - http://neo4j.com
[5] RIPE Atlas https://atlas.ripe.net