Assignment 5: Explore CAIDA's Internet Topology Data Kit (22pts)

Due on Wednesday March 22, 2023 11:59pm

1 Introduction

The goal of this assignment is to become familiar with CAIDA's Macroscopic Internet Topology Datakit (ITDK), which contains heavily-annotated router-level topologies collected every 6-12 months beginning 2010.

A router is a physical device that transports packets between Internet endpoints by forwarding packets on links between it and other routers. Each router has multiple *interfaces*, each of which connects to at least one other router. Operators of these routers configure each one of these interfaces with an *IP address*, and therefore each router has multiple IP addresses.

CAIDA infers which IP addresses belong to the same router using heuristic methods *MIDAR* [1] and *iffinder* [2]. These methods send probe packets to router IP addresses, and infer which IP addresses belong to the same router based on the responses they receive. CAIDA annotates these routers, as follows:

1. **Hostnames:** the hostname associated with a router IP address, such as `dc-sdsc-100ge-sdg-agg4.cenic.net` for 137.164.23.43, recorded in the DNS.

2. **AS Number:** the network, identified by ASN, inferred by *bdrmapIT* [3] to operate the router. *bdrmapIT* inferred that AS195 (SDSC, UCSD) operated the router with IP address 137.164.23.43.

3. **Geographic Location:** the location of the router, inferred using one of three methods:

   (a) **Hoiho**, which uses information encoded in hostnames to infer geographic location [4]. For example, `sdg` in the hostname above means "San Diego".

   (b) **IX**, which uses geographic information reported by network operators in *PeeringDB* [5] to identify the location of a router based on its presence at an IXP.

   (c) **Maxmind**, a commercial provider of geolocation information that also provides a coarse-grained geolocation database for free [6].

4. **Links:** the links attached to the router, as observed by *traceroute*.

The most recent ITDK built by CAIDA was in February 2022 (2022-02). The ITDK is accessible via `https://www.caida.org/catalog/datasets/internet-topology-data-kit/`.

That page also contains a description of the dataset, which is provided as a series of individual files. For this assignment, we have pre-processed the individual files for the 2022-02 ITDK into a single SQLite database. We have indexed the database to minimize query response time, and have provided you simple perl scripts to query the database. You may use these scripts in your analyses, or create your own script to conduct analyses, perhaps using the perl script as documentation.

1.1 Database, Scripts and API

The database itself is available at `https://www.caida.org/~mjl/tmp/202202.sqlite`. You can use the `sqlite3` tool to explore the schema of the database, as follows:

```
$ sqlite3 202202.sqlite
SQLite version 3.37.2 2022-01-06 13:25:41
Enter ".help" for usage hints.
sqlite> .schema
CREATE TABLE IF NOT EXISTS "nodes" ("id" int not null,
   "asn" int, "geo_method" int, "geo_lat" real, "geo_lng" real,
    geo_continent string, geo_country string, geo_region string,
    geo_city string);
CREATE TABLE IF NOT EXISTS "ips" ("id" int not null,
   "node_id" int not null, "addr" string not null, "name" string);
CREATE TABLE IF NOT EXISTS "links" ("id" int not null, "node_id" int);
CREATE INDEX nodes_id on nodes(id);
CREATE INDEX ips_node_id on ips(node_id);
CREATE INDEX ips_addr on ips(addr);
CREATE INDEX links_node_id on links(node_id);
CREATE INDEX links_id on links(id);
CREATE INDEX nodes_asn on nodes(asn);
```

The database is an 800MB file, and is intended to be read by a script.

The query script is available at `https://www.caida.org/~mjl/tmp/db-itdk-query.pl`. We have provided a third script to calculate the distance between two lat/long values, at `https://www.caida.org/~mjl/tmp/distance.pl`. The query script relies on two external perl modules: DBI, and DBD::SQLite. If you are running Ubuntu, you can install these with `apt install libdbd-sqlite3-perl`. Because the script can return quite a bit of information, you should pipe the output through `less` or `more`, e.g.
`perl db-itdk-query.pl 202202.sqlite AS701 | less`

To save you from having to install the software, we have combined all functionality inside one convenient web interface:

`http://nids.caida.org:55555/.`

(If your browser automatically converts to https, you will have to manually type http into the URL.) Enter the parameters inside the input field like you would in the command line.

For example, in the last example a above, enter `AS701` in the field.

Looking at the second entry that that query returns:

```
1  # id: 7034
2  # as: 701
3  # geo: US-GA Atlanta 33.6367,-84.428101 hoiho
4  204.148.224.41
5  204.255.168.253 0.ae10.BR3.ATL4.ALTER.NET
6  140.222.225.169 0.ae2.BR3.ATL4.ALTER.NET
7  67.111.23.190 67.111.23.190.ptr.us.xo.net
8  65.195.233.33 N-A.4-1-0.BR3.ATL4.ALTER.NET
9  4.68.71.62 Verizon-level3-Atlanta2.Level3.net
```

1. The first line reports the router's identifer – assigned sequentially by CAIDA and recorded in the ITDK.
2. The second line reports the AS inferred by bdrmapIT to operate the router.
3. The third line reports the geographic location recorded in the ITDK. This line reports two-level country and state codes (US-GA), a city location (Atlanta), latitude and longitude, and the method that reported this location.
4. The fourth line reports an IP address on the router, which in this case did not have an associated hostname.
5. The fifth line reports an IP address on the router, and the associated hostname. The hostname reports the link is "Aggregated Ethernet" – `ae`, that the interface is on a "Border Router" – `br`, and is located in Atlanta – based on the IATA airport code (`atl`) for Atlanta. The IP address was allocated to alter.net, which is a suffix associated with Verizon.
6. The seventh line reports a hostname associated with xo.net, a suffix associated with XO networks.
7. The ninth line reports a hostname associated with level3.net, where Level3 is reporting that the address is used on an interconnection it has with Verizon.

The script supports the following queries:

- **Identify Border Routers Between ASes:**
  `perl db-itdk-query.pl 202202.sqlite xxx:yyy`
  queries the database for all router-level links between AS xxx and AS yyy.
  Each router-level link is separated with a line formatted as follows:
  `## xxx:Naaa yyy:Nbbb cccc`
  where xxx and yyy are the ASNs in the query, aaa and bbb are unique router IDs, and cccc is a unique link ID. If the ASes have multiple router-level links between them, each link will begin with this string. The routers are formatted as above.

- **Identify Routers Operated by an AS:**
  `perl db-itdk-query.pl 202202.sqlite ASxxx`
  queries the database for all routers inferred to be operated by AS xxx. The routers are formatted as above.

- **Identify Router by ID:**
  `perl db-itdk-query.pl 202202.sqlite Naaa`
  queries the database for a router with ID aaa. The router is formatted as above.

- **Identify Router by IP Address:**
  `perl db-itdk-query.pl 202202.sqlite 132.239.254.161`
  queries the database for a router with the given IP address. The router is formatted as above.

If you pass `-json` to the script, the query script will return lines of JSON objects.

Finally, we have combined all functionality inside one convenient web interface: `http://nids.caida.org:55555/`.

Enter the parameters inside the input field like you would in the command line. For example, in the last example above, enter the IP address `132.239.254.161` in thhe field.

1.2 Due Date

Wednesday March 22, 2023 11:59pm

1.3 Submission Instructions

You will answer the questions in a written PDF report and submit to Gradescope.

2 Questions / Tasks

1. Query the database for all router-level links inferred between Level3 (AS3356, a Tier-1 transit provider) and Netflix (AS2906, a content provider).

   (a) Border routers tend to be geographically adjacent. How many links between border routers indicated that the routers were geographically adjacent, defined as located within the same city, or no further than 40km apart? Report the link IDs involved. How many routers were not inferred to be geographically adjacent? Report the link IDs involved. (2pts)

   (b) Excluding the border router instances that appear to not be geographically adjacent, how many different locations did Netflix and Level3 appear to peer? Produce a table with the inferred locations. (2pts)

   (c) Write a paragraph summarizing *your* interpretation of these findings. Focus on broad themes, e.g., economics, performance, rather than interconnection locations. (3pts)

2. Query the database for all routers inferred to be operated by China Unicom (AS4837).

   (a) Four routers (IDs 200, 243, 267, 316) are located on the U.S. West Coast (two each in Los Angeles and San Jose) and connect with other ASes for transit. You can use the hostnames of the router interfaces to help you identify the ASes involved, as well as the ASN originating the longest matching prefix (using CAIDA's Prefix to AS mapping dataset for 25th of February 2022 [7]). Produce a table that identifies the networks/ASes that interconnect with each of these routers. (2pts)

   (b) Write a paragraph summarizing your interpretation of the table. Why would China Unicom connect with the same ASes at the same location more than once? Why would China Unicom connect with the same networks/ASes at different locations? (3pts)

3. Query the database for all router-level links inferred between Limelight Networks (AS22822, a content distribution network) and Arelion (AS1299, a Tier-1 transit provider).

   (a) Router 3358 appears to have one interface with a stale hostname, because the operator-annotated geolocation disagrees with other geolocation hints. Which hostname is stale? Explain your answer. (2pts)

   (b) Some links had AS22822 routers geolocated with Hoiho, while some AS1299 routers were geolocated with Maxmind, and the inferred geolocation rarely agrees. Where are AS1299 routers typically geolocated with Maxmind? Why do you think that is? Explain your answer. (3pts)

4. Query the database for all router-level links between any two of the following ASes: AS174 (Cogent), AS701 (Verizon), AS714 (Apple), AS1299 (Arelion), AS2906 (Netflix), AS3257 (GTT), AS3320 (Deutsche Telekom), AS3491 (PCCW), AS5511 (Orange), AS6453 (TATA), AS6461 (Zayo), AS6762 (Telecom Italia), AS6830 (Liberty Global), AS12956 (Telefonica), AS13335 (Cloudflare), AS15133 (Edgecast), AS15169 (Google),

AS20940 (Akamai), AS22822 (Limelight). Write two paragraphs explaining what you observe between the ASes. If you are unsure about what you are seeing, send an email to Matthew for advice (mjl@caida.org). (5pts)

References

[1] Ken Keys, Young Hyun, Matthew Luckie, and k claffy. Internet-scale IPv4 alias resolution with MIDAR. *IEEE Transactions on Networking*, 21(2):383–399, April 2013.

[2] Ken Keys. Internet-scale IP alias resolution techniques. *CCR*, 40(1):50–55, January 2010.

[3] Alexander Marder, Matthew Luckie, Amogh Dhamdhere, Bradley Huffaker, k claffy, and Jonathan M. Smith. Pushing the boundaries with bdrmapIT: Mapping router ownership at Internet scale. In *Proceedings of the 18th ACM SIGCOMM Conference on Internet Measurement (IMC)*, pages 56–69, October 2018.

[4] Matthew Luckie, Bradley Huffaker, Alexander Marder, Zachary Bischof, Marianne Fletcher, and k claffy. Learning to extract geographic information from Internet router hostnames. In *Proceedings of the 17th ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, December 2021.

[5] Peeringdb. `https://www.peeringdb.com/`.

[6] Maxmind. `https://www.maxmind.com/`.

[7] CAIDA prefix2as for 25th of February 2022. `https://publicdata.caida.org/datasets/routing/routeviews-prefix2as/2022/02/routeviews-rv2-20220225-0600.pfx2as.gz`.