Programming in Java
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: 320341
Date: 2015-11-08
Due: 2015-11-16

Problem Sheet #4

Problem 4.1: *augmenting and grouping network flow records* (4+3+2+4+2 = 15 points)

In this assignment, you are going to further extend the network flow processing program that you have written for the previous assignment. The following extensions should be implemented:

a) The Internet consists of a set of inter-connected "Autonomous Systems". An Autonomous Systems (AS) is a set of routers and networks under the same administration. An AS is uniquely identified by its AS number. Global IP address blocks are assigned in the form of IP address prefixes to Autonomous Systems.

   Implement a class that represents information about an Autonomous System. Given an IP address, it should be possible to obtain an instance of your class representing an Autonomous System. An object of your class should understand methods to obtain the AS number, the AS holder information, and the prefix matching the address given. The necessary information can be obtained by sending a request to the following REST service:

   `https://stat.ripe.net/data/prefix-overview/data.json`

   Assuming the IP addresses 192.0.2.1 or 2001:db8::1, lookup requests would be:

   `https://stat.ripe.net/data/prefix-overview/data.json?resource=192.0.2.0`

   `https://stat.ripe.net/data/prefix-overview/data.json?resource=2001:db8::1`

   (Note that the example IP addresses using in this writeup are not assigned to any ASes and hence you may want to try this with different addresses.)

   Your implementation must make sure that data is cached. If I obtain the AS object for a given IP address multiple times, only one REST call should be made. The parsing of the JSON response returned by the REST service can be implemented by making use of the GSON library.

b) Extend your flow processing program with a mechanism that allows flow records to be annotated. Use this mechanism to augment flow records with their source and destination AS number. You may also consider to annotate flow records with AS objects instead of just AS numbers.

c) Define an interface `FlowGrouper` that can be used to group flow records that have a common property. This is achieved by inspecting properties of a flow records and/or its annotation and returning a key that identifies the group the flow belongs to. The key may be represented as a simple string.

d) Extend your network flow processing program such that an application can request statistics for flow record groups. Test this my producing statistics for all source AS / destination AS pairs observed in a given network flow record trace.

e) Use the `maven` build system to structure your project. Write unit tests for your new or extended classes to demonstrate that your classes are working as expected.