Network Programming Laboratory

27 June 2022

Exercise

We aim at implementing a simple monitoring system for TCP flows which consists of a centralized *collector* (client) and multiple *probes* (clients) to be deployed at possibly different vantage points.

Traffic probe. By using the callback mechanism provided by the `pcap` library, the probe captures only TCP packets form a network interface/file (specified from the command line). For every packet carrying a TCP segment, the probe:

- collects the header field values: Source IP, Destination IP, Source Port, Destination Port, Segment Length
- prepares a report object with fields: `src_ip, dst_ip, src_prt, dst_prt, length`
- encodes the above object by means of JSON, with the corresponding keys
- serializes the JSON object and sends it to a UDP collector server listening on a given IP address and port (arbitrarily set to) 12000.

Data collector. The data collector receives the reports from the probes on UDP port 12000 and every 20 seconds prints on screen *the top five TCP hitters*, defined as the 4-tuples (Source IP, Destination IP, Source Port, Destination Port) that carried the highest volume of traffic (TCP net data only) in the last observation window (of 20 seconds).

1. Write a C++ program that implements the traffic probe.
2. Write a C++ multi-threaded program that implements the data collector.

Note: all library functions from the NPL repository developed in the course can be used, as well as your favorite external JSON library (e.g., the `nlohmann` library).