TCP/IP Communication Flows into Sentence-like Transcriptions

Allan Kálnay

November 28, 2020

The overall goal of the project is to design a suitable symbolic schema that transforms TCP/IP flows into sentence-like transcriptions and evaluate suitability for knowledge discovery and (ideally) attack detection.

Network traffic data that are used as an input for this project are in .pcap format. In order to analyze these data, we have to extract features from them. We have several ways to do this from the basic tools like *tshark*[1] - a network traffic analyzer - to more sophisticated ones like *go-flows*[2] or our own scripts that may be constructed in case of more complex feature extraction.

Next key part of the project is proposing a map of key symbols that will be used for transforming the network traffic into. We need to construct a text-like transcriptions that will be suitable for non-encrypted as well as encrypted communications. Let's preliminary call this text-like transcription NTFT (Network Traffic Flows Transcriptions). For this purpose, it is needed to make a set of scripts or a pipeline of scripts for transforming .pcap files into NTFT.

As already mentioned above, we want to have our transcriptions suitable also for encrypted communications. This makes us limited at feature extraction, since we have to extract such features that are available both for encrypted and non-encrypted communications.

The processing scripts for transforming .pcap files into NTFT are required to be written in Python. This programming language choice is very handy due to its large community and popularity that was raising a lot last years[3].

For the testing purposes we will use the following network traffic data – the CIC2017 IDS dataset[4] (synthetic, labeled) and MAWI WIDE datasets[5] (real, unlabeled).

The student is supposed to explore the literature related to the topic, so that he can theoretically disclose what are network traffic features able to discriminate about traffic types.

The student should analyze traffic transcriptions and evaluate their performances. This can be done by comparing models learned on the data output of this project with models learned on baseline scheme that uses the same or very similar features that are expressed as numerical vectors.

The final step is to extract knowledge from the discovered patterns and provided analysis.

[1] https://www.wireshark.org/docs/man-pages/tshark.html

[2] https://github.com/CN-TU/go-flows

[3] https://www.kdnuggets.com/2019/05/poll-top-data-science-machine-learning-platforms.html

[4] https://www.unb.ca/cic/datasets/ids-2017.html

[5] https://mawi.wide.ad.jp/mawi/