A Traffic Merging and Generation Framework for Realistic Synthesis of Network Traffic

Cem Gündogan, Sandro Passarelli, Peter Hillmann, Christian Dietz, and Lars Stiemert

Universität der Bundeswehr München
85579 Neubiberg, Germany
{Cem.Guendogan, Sandro.Passarelli, Peter.Hillmann, Christian.Dietz, Lars.Stiemert}
@unibw-muenchen.de

Introduction:
The Internet is steadily growing and is of increasing importance for our economy and society. Due to this increased importance it is also in the focus of attacks, e.g. distributed denial of service (DDoS) attacks. As attackers dynamically change their attack behaviour, novel detection approaches that are able to automatically adjust to these dynamic attacks are needed. To train and test such network anomaly detection systems, it is necessary to provide realistic data. As of today, this area of research suffers from the lack of publicly available datasets that can be used to train and test anomaly detection systems and are exchangeable to allow reproducible research. Therefore, we propose a novel framework that enables researchers and developers to generate customizable synthetic datasets. It not only allows to generate fully-synthetic network traffic, but also to generate semi-synthetic network traffic by merging of multiple network captures from real-live environments. Further, it allows the mapping of IP addresses as well as the modification of other header fields, if desired. This enables researchers and developers to exchange network traces from sensitive environments without revealing any sensitive end-user related information, while perceiving the relevant characteristics of the network(s) and attack(s). In the following, we provide a description of, the problem, our concept and the features of our solution, the architecture and functional model and finally provide a short summary together with an outlook for future work.

Problem:
Testing of IDS and IPS is often suffers from the lack of available ground truth data sets that are derived from rela-life environments and are publicly available. Such ground truth data includes labels for each sample. Based on these labelled samples one can analyse the accuracy of a detection system. Due to privacy constraints and the overhead to derive sufficient ground truth datasets, such data sets are usually either shared under non disclosure agreements or not at all. Thus, many researchers create or use synthetic datasets to make their research reproducible, even though it is known that synthetic data can easily lead to false conclusions.

Concept and Functionality:
Our framework provides a novel approach, by combining the benefits of real live data captures and synthetic data generation.It provides multiple ways to generate and manipulate network traffic captures, one of which is a simple, random based generation. Users are able to specify the generation process with parameters, i.e. source and destination IP-addresses and ports, amount and types of headers and payload. Additionally, it is possible to produce, e.g., uncompleted TCP handshakes and allows to perform packet manipulations across dataset based on user defined distributions.
As Figure 1 shows, users can also merge random generated and live recorded traffic to ensure an

even better, more realistic outcome. The merging will automatically keep any given uncompleted handshakes in order, private addresses will be mapped and masked with a user specified super IP.

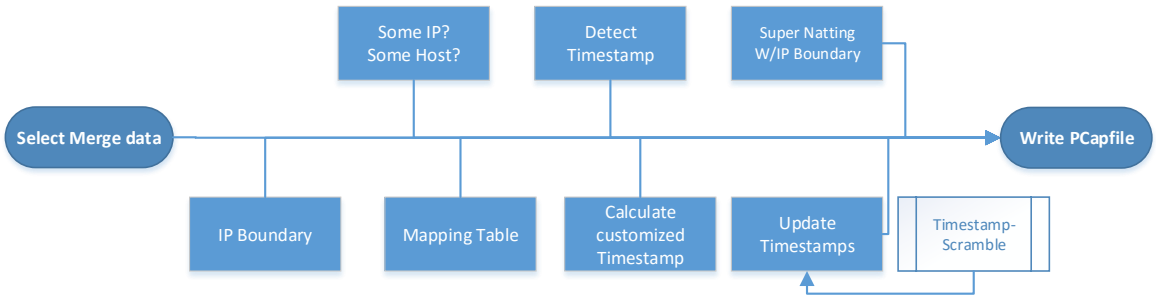

Figure 1: *PCap merging*

Model and Implementation:
First, to generate network traffic with user defined parameters, our framework displays the first rudiments. Secondly, the it requires one ore multiple PCap file as input for the merging process. Our framework also provides an application programming interface to read and generate network traffic and makes use of the open source java library jNetPcap. Figure 2 describes the different options to dump an output PCap file. The first option uses PCap-Templates for replacing or adding more packet information, like which protocol is used for packet switching. Currently, our reference implementation supports the UDP- and TCP-protocol. When the user selected TCP and defined a quantity of TCP packets with completed and uncompleted TCP 3-Way-Handshakes, our framework can for example reorder these packets based on the user definition. A distribution function handles the network data dissemination across the final output file. To build up a PCap it is essential to check the headers and checksum as last step before dumping generated PCaps to disk.

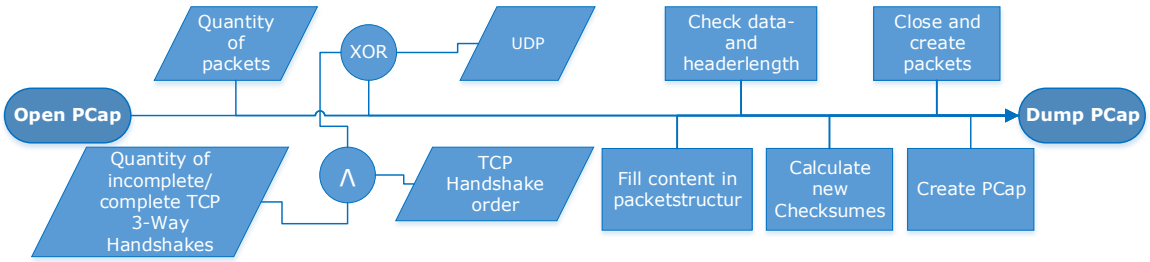

Figure 2: *Creating PCap*

Conclusion and Future Work
The whole framework is planned to be open source and available for the community. Features like a build in traffic transmission to instantly test any IDS and IPS are included. The validation of our approach is still in its initial state and planned to be extended in future work. After a successful validation we plan to implement the support of more different network traffic content by adding more protocols. Furthermore, we connect the databases of IDS and IPS with signatures of attacks and use this information to include packets based on attack patterns into the generated Pcap.