BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2023
http://excel.fit.vutbr.cz

Traffic Capture Infrastructure

Bc. Lukáš Hejcman*

Abstract

Diverse and high-quality datasets are the cornerstone of any machine learning algorithm development and research. However, creating such datasets in the context of modern high speed, distributed, and privacy sensitive networks can be problematic. This is why we designed and implemented a solution in the form of Traffic Capture Infrastructure; a one-stop system for creating, processing, and handling large datasets created from network traffic across a large number of network nodes. Furthermore, our system supports extensive user management features to ensure dataset privacy and system integrity. In this paper we present the architecture and inner workings of this system. We present its advantages and possible improvements. Lastly, we prove the value of this system with a number of publications that have used our system for creating their underlying dataset.

*xhejcm01@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

The quantity and quality of underlying datasets is seen to be one of the main limiting factors in the development of new machine learning algorithms [1]. However, data processing can also be very time consuming and is widely seen as a one of the least enjoyable tasks in data science [2]. Furthermore, the difficulty of preserving data privacy for sensitive datasets is self evident from the recent increase in the interest in the methods of collaborative learning, which are designed to allow training models without sharing their underlying dataset [3].

In this work, we implemented a system for network dataset creation called Traffic Capture Infrastructure (TCI). This system is designed to be a one stop solution for creating datasets of network traffic at a large scale, including their capture, processing, user management, and more. This system has already been deployed on the infrastructure of CESNET (Czech Education and Scientific NETwork), a developer and operator of national e-infrastructure for science, research, development and education in the Czech Republic. The system has been used successfully as a part of multiple papers, publications, and open-source datasets.

2. System overview

The Traffic Capture Infrastructure (TCI) system takes inspiration from the existing network flow systems and is composed of multiple interconnected modules; an overview of which can be seen in the System Architecture section of the accompanying poster. The traffic capture itself is managed by the TCI hive and the TCI drones, which work similarly to a flow collector and exporter respectively. This backend captures the traffic using 1 to n drones. The captured data is then collected and merged using the TCI hive.

The whole backend system can be controlled and managed using the other distinct part of the TCI system, the Command Line Interface (CLI) or the Information System (IS). The CLI allows for a low level access to the TCI system, which can be used for initial setup, administrator management, or in environments where no further functionality is required from the TCI system. On the other hand, the information system makes TCI a system which can be comfortably deployed and integrated into existing infrastructure in an academic or a corporate environment, where privacy and integrity considerations arise.

The whole TCI system is written in Python using the Flask [4] framework. Flask was selected due to its lightweight size and large number of extension libraries. The system uses SQLAlchemy [5] as an ORM which

gives us the flexibility of deploying the system using a large number of available database backends. The web interface was built using Angular [6].

3. Real world usecases and results

The described TCI system has been deployed on the infrastructure of CESNET. This long-term deployment has been in operation successfully for multiple months without any major downtime, and has been used for creating hundreds of capture jobs. The deployed instance of the TCI system takes full advantage of the developed Information System to enable integration into the CESNET LDAP system for authorizing user access and their privileges within the system. This deployed system has already been used for multiple publications.

Luxemburk and Čejka used this system for capturing a large number of TLS messages, which were used for annotating existing flows with a their Server Name Indicator [7]. This dataset was then used to implement and train a classifier which achieved a 97.04% classification accuracy and detected 91.94% of unknown services with a 5% false positive rate.

Luxemburk et al. again used the TCI system for annotating an existing dataset of millions of traffic flows with a packet-level feature set [8]. This dataset was then used for training a machine learning classifier with results that surpass the state-of-the-art solutions, and that has been shown to be viable for production deployment.

Tropková et al. used this system to develop and implement a new network traffic characteristic called Sequence of packet Burst Length and Time (SBLT) [9]. The value of this approach was shown using an implemented classifier with an achieved accuracy of 99%. The used dataset is also publicly available [10].

Smejkal used this system for creating a detailed dataset of SSH connections used for classification and user identity identification [11]. This dataset was then used for creating a classifier of SSH connections with an accuracy of up to 99.78%.

The TCI system is also being used by CESNET-CERTS, a computer security incident response team, in cooperation with CESNET network operators [12]. This team uses TCI to capture samples of Distributed Denial of Service (DDoS) data in order to increase the accuracy of existing attack mitigation and filtering rules.

Furthermore, Jeřabek et al. created a public dataset of DNS over HTTPS traffic [13]. This dataset, amongst other datasets being created by the TCI system, is being used for the development of future publications.

4. Future work

The main improvements that are planned for the TCI system are centered around user comfort and system autonomy. The main drawback of the TCI system is currently the fact that users must manually transfer the created and processed dataset to the server where further development will take place. The proposed solution to this problem is to allow users to automatically set a destination for each capture job. After processing, the TCI system will connect to the desired destination using a user specified SSH key and transfer the data.

Another perceived problem is regarding the deployment of the TCI system. From an administrator point of view, it would be useful to include storage management into the TCI system. The administrator could set data usage quotas to different users, who could not capture jobs larger than a certain threshold.

5. Conclusion

In this paper we presented an existing Traffic Capture Infrastructure system. This system was created as a one-stop solution to creating and processing datasets of network traffic for researchers and administrators of computer networks. We presented an overview of the structure and architecture of the system, and explained its main functionality. We gave an overview of the main usecases of the system, and proved its usefulness using a list of publications which used the TCI system during their development.

References

[1] Alon Halevy, Peter Norvig, and Fernando Pereira. The Unreasonable Effectiveness of Data. 24(2):8–12.

[2] Gil Press. Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says.

[3] Balazs Pejo, Qiang Tang, and Gergely Biczok. Together or Alone: The Price of Privacy in Collaborative Learning.

[4] Welcome to Flask — Flask Documentation (2.1.x).

[5] SQLAlchemy - The Database Toolkit for Python.

[6] Angular Components Team. Angular Material.

[7] Jan Luxemburk and Tomáš Čejka. Fine-grained TLS Services Classification with Reject Option.

[8] Jan Luxemburk, Karel Hynek, and Tomáš Čejka. Detection of HTTPS Brute-Force Attacks with Packet-Level Feature Set. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0114–0122.

[9] Zdena Tropková, Karel Hynek, and Tomáš Čejka. Novel HTTPS classifier driven by packet bursts, flows, and machine learning. In *2021 17th International Conference on Network and Service Management (CNSM)*, pages 345–349.

[10] Zdena Tropková, Karel Hynek, and Tomáš Čejka. Dataset used for HTTPS traffic classification using packet burst statistics.

[11] Radek Smejkal. Klasifikace komunikace SSH protokolu.

[12] Trusted Introducer : Directory : CESNET-CERTS.

[13] Kamil Jeřábek, Karel Hynek, Tomáš Čejka, and Ondřej Ryšavý. DNS over HTTPS - Real World Dataset.