

Trace-Share: Towards Provable Network Traffic Measurement and Analysis

Special Session on Network Security at Prague Embedded Systems Workshop (PESW 2019)

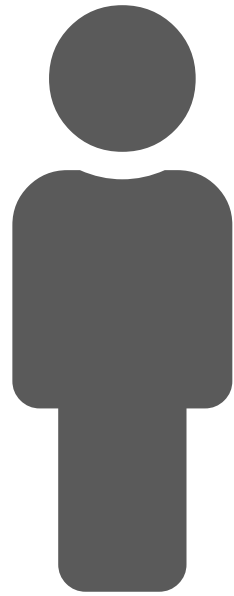
June 28, 2019

Milan Cermak

Institute of Computer Science, Masaryk University, Brno



CSIRT-MU







Issues of Network Traffic Analysis Research

challenges that everyone must deal with

Lack of **research standards**

- missing rules for research data collection, analysis, sharing, and ethics of their usage

Inaccessibility of **appropriate datasets**

- real-world data cannot be reliable annotated and needs to be anonymized, artificial data are not sufficiently realistic and provides a limited set of events in network traffic

Inability to **prove research** results

- we have no approach to assess the proposed analytical method reliably

No **verification** of other researchers' findings

- data and algorithms are kept in private which leads to the impossibility of research reproducibility



The Initial Idea

what we realized during our research

Full packet capture of a **single event** can be publicly shared – one network event contains only a minimum of personal data and can be publicly shared and annotated

Packet capture can be „simply“ **transformed** – packet fields can be changed to predefined values and adapted according to real-world data

Events can be **mixed** with each other or with real-world data – we have access to the real-world data, but we need an annotation or a ground truth

Trace-Share: Towards Provable Network Traffic Measurement and Analysis

our goal is to cover all issues related to research provability and dataset usage, but we need to start from the beginning...

Annotated Unit

single event in network traffic that is normalized and annotated



Unit of network traffic

- A single complex event in a network containing all connections and packets related to the event
- Full packet capture with all application data (Pcap or PcapNg)
- Known capture environment and all characteristics of the network

Normalized unit

- Unification of the unit to simplify further processing of all events
- MAC addresses rewritten to 00:00:00:00:00:01 (source), 01:00:00:00:00:01 (destination)
- IP addresses rewritten to 240.0.0.2 (source), 240.125.0.2 (destination)
- Capture start set to zero epoch time

Annotated unit

- Normalized unit enriched to its annotation
- Capture properties, event description, and optional tags (e.g., MITRE ATT&CK™ classes)

Annotated Unit of SSH Dictionary Attack

theory is nice but real example is better

- Various tools providing **a lot of options** results in multiple annotated units for each variant
- **Successful and unsuccessful** attacks can form different annotated units
- Required number of connections is not specified, **you decide** what an attack is

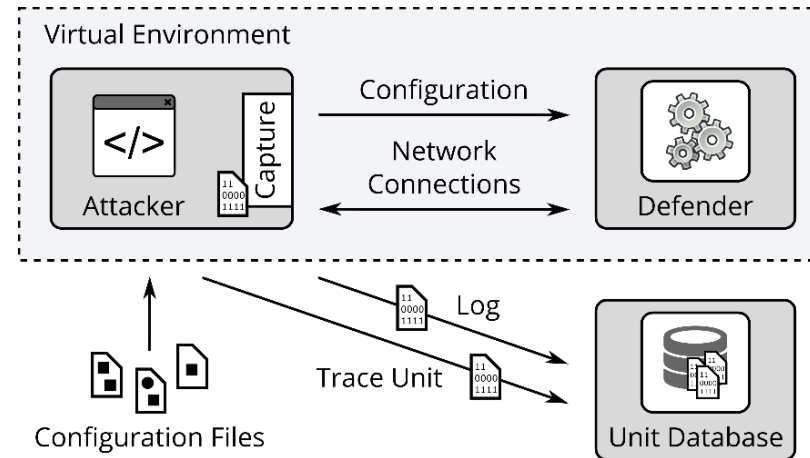
of a reassembled PDU]

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
240.0.3.4	41024	240.125.0.2	22	45	7149	23	4114	22	3035	0.000000	10.9295	3011	2221
240.0.3.4	41026	240.125.0.2	22	37	6557	16	3588	21	2969	10.923998	9.4027	3052	2526
240.0.3.4	41028	240.125.0.2	22	37	6493	16	3524	21	2969	19.945335	9.5902	2939	2476
240.0.3.4	41030	240.125.0.2	22	39	6561	18	3592	21	2969	24.946822	9.5409	3011	2489
240.0.3.4	41032	240.125.0.2	22	37	6621	16	3652	21	2969	34.946551	10.3365	2826	2297

https://github.com/CSIRT-MU/Trace-Share/tree/master/datasets/SSH_dictionary_attacks

Automated Creation of Annotated Units

a simple way to obtain all variants of the desired event



- **Virtual environment** orchestrated by Vagrant and Ansible
- Configurable **management script** deployed on the Attacker able to manipulate settings of used hosts, run given commands, and start captures of all related network traffic
- **Full packet trace** is generated for all given commands
- Publicly available at <https://github.com/CSIRT-MU/Trace-Share/tree/master/trace-creator>

Challenges of Annotated Units

besides benefits, there are still issues that need to be addressed



- **No sensitive content** of a traffic
- Accurate **annotation**
- Easily accessible **data recency**



- **Variability** of network environment
- **Normalization** in application data
- **Annotation** format

Semi-labeled Dataset

combination of annotated units with real-world network traffic

Semi-labeled dataset = additional of ground truth baseline in your unlabeled real-world data via injection of selected annotated units

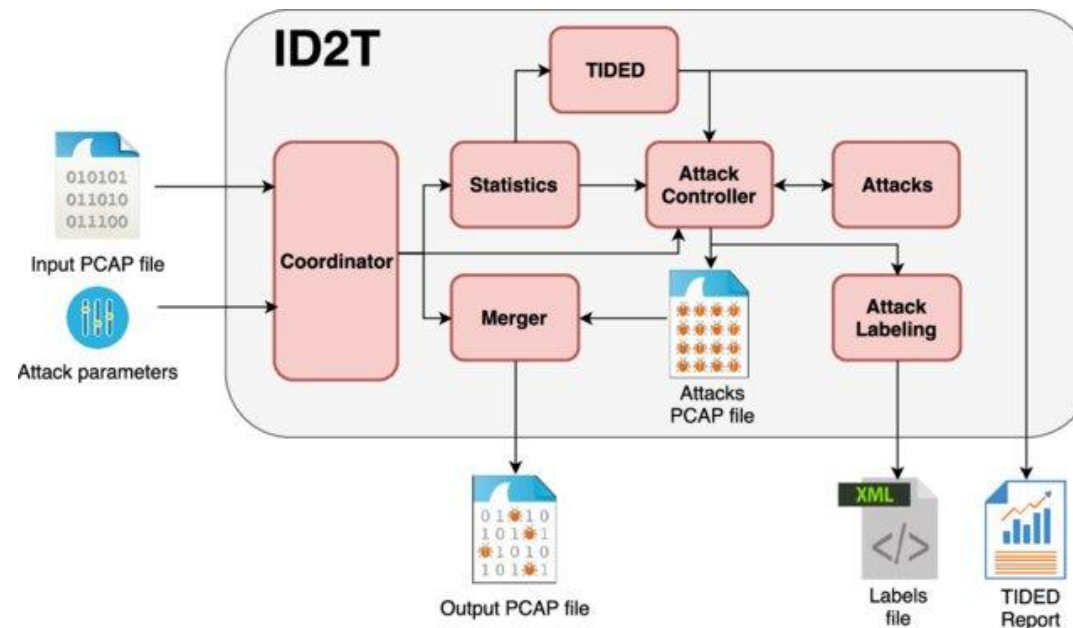
1. **Select** annotated units based on your interest
2. **Capture** real-world network traffic within your environment
3. **Compute** characteristics of the real-world traffic capture
4. **Modify** annotated units to reflect characteristics of the real-world traffic
5. **Merge** annotated units and real-world traffic capture

Intrusion Detection Dataset Toolkit (ID2T)

a tool with awesome features suitable for our goal

"ID2T facilitates the creation of labeled datasets by injecting synthetic attacks into background traffic injected synthetic attacks blend themselves with the background traffic by mimicking the background traffic's properties to eliminate any trace of ID2T's usage."

Publicly available at <https://github.com/tklab-tud/ID2T>



Trace-Share: ID2T

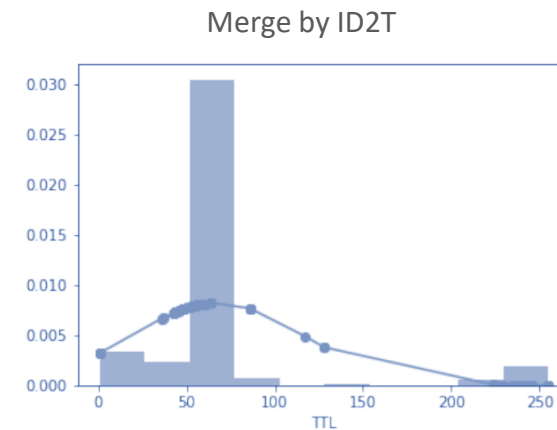
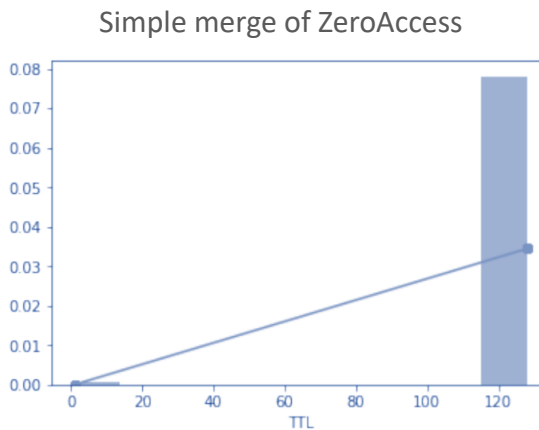
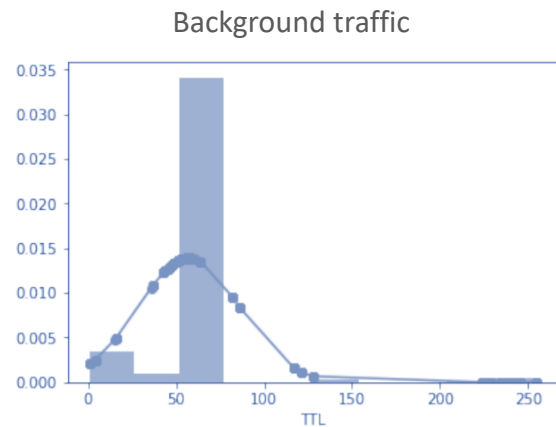
an extension providing injection of existing packet traces

Extension of the Attack Controller to support usage of existing packet traces

- Instead of prescription for a synthetic attack, you can provide annotated units and specify packet fields that should be adapted according to the background traffic

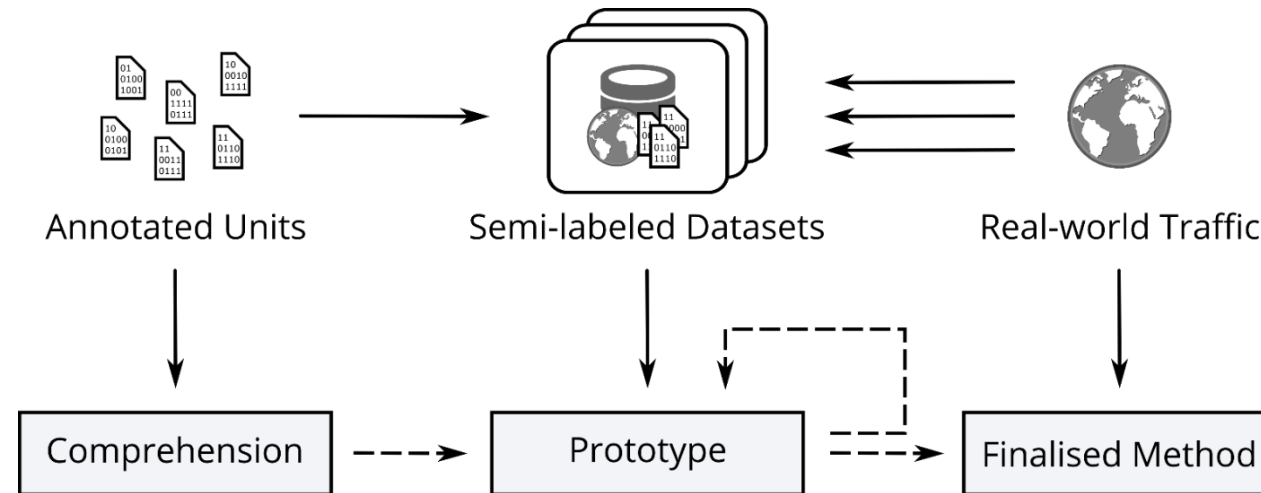
Adaptation of variable packet fields in major network protocols

- MAC and IP addresses for ARP, IPv4, IPv6, ICMPv4, ICMPv6, DNS, HTTPv1
- Ports in UDP and ports, window size, maximum segment size, time-to-live in TCP
- Packet timestamp



Analysis Development with Semi-Labeled Datasets

usage example of annotated units and semi-labeled datasets

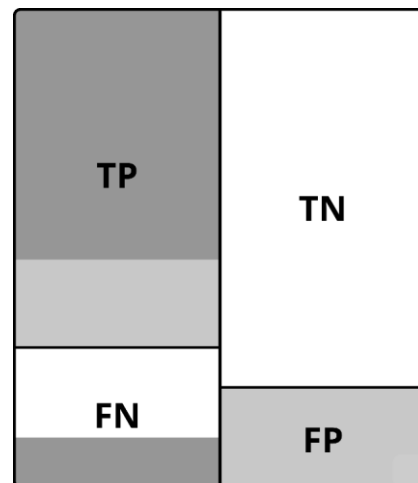


1. Use annotated units for **an initial comprehension** of network traffic related to your problem
2. **Enrich your real-world data** with selected annotated units and prepare semi-labeled datasets
3. Train and develop the analysis prototype **using baseline** provided by generated datasets
4. Finalize the method after you **can recognize all** desired annotated units

Analysis Evaluation Using Semi-Labeled Dataset

injected labels serves as a ground truth in unlabeled data

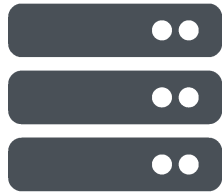
- Injected annotated units serves as a **ground truth baseline** in unlabeled dataset
- **Balanced** quantitative (objective metrics) and qualitative (real-world data characteristics) aspects
- Unknown positives need to be **verified manually** and shared



- Annotated units
- Other identified events
- Uncertainty

Data Sharing Platform

generate your units, share them, and use what others have created



- **Community** hub
- **Storage and management** of annotated units
- **Assisted** uploading, normalization, annotation, and adaptation of annotated units




- Inspired by OpenML platform (see <https://openml.org>)
- Prototype available at the end of the year (see <https://github.com/Trace-Share>)

Summary

what you should take away from this presentation

- You don't need to share the entire network traffic, **share only selected events!**
- **Mix events** between themselves and with real-world traffic
- **Share your differences** and provide your annotated units to others
- **Prove** your research results!
- Check our repository <https://github.com/Trace-Share>
- If you are interested in this topic, contact me at cermak@ics.muni.cz

trace  **share**
by CSIRT-MU

Prove your research by shared trace!

 <https://github.com/Trace-Share>

 @csirtmu

Milan Cermak

cermak@ics.muni.cz

MUNI



EUROPEAN UNION
European Structural and Investment Funds
Operational Programme Research,
Development and Education

