Cyber Situation Awareness via IP Flow Monitoring

Tomas Jirsik, Pavel Celeda
Institute of Computer Science
Masaryk University
Brno, Czech Republic
{jirsik,celeda}@ics.muni.cz

***Abstract*—Cyber situation awareness has been recognized as a vital requirement for effective cyber defense. Cyber situation awareness allows cybersecurity operators to identify, understand, and anticipate incoming threats. Achieving and maintaining the cyber situation awareness is a challenging task given the continuous evolution of the computer networks, increasing volume and speeds of the data in a network, and rising number of threats to network security. Our work contributes to the continuous evolution of cyber situation awareness by the research of novel approaches to the perception and comprehension of a computer network. We concentrate our research efforts on the domain of IP flow network monitoring. We propose improvements to the IP flow monitoring techniques that enable the enhanced perception of a computer network. Further, we conduct detailed analyses of network traffic, which allows for an in-depth understanding of host behavior in a computer network. Last but not least, we propose a novel approach to IP flow network monitoring that enables real-time cyber situation awareness.**

***Index Terms*—cyber situation awareness, IP flow, monitoring, network, real-time, host identification**

I. Introduction

Computer networks have become an inherent part of our everyday life. They facilitate a variety of services, including the ones critical for society. As society has become increasingly dependent on these services, computer networks have become lucrative targets for various attackers and fraudsters. The protection of computer networks from harm caused by attackers has become an important topic. An essential prerequisite for the adequate protection of a computer network is an ability to perceive the processes in the network, comprehend their meaning and relations, and predict the future state of a network – in other words, to gain a *cyber situation awareness*.

Cyber situation awareness is a concept that supports cybersecurity operations in cyber defense. The goal of this concept is to provide an operator with sufficient information that enables him/her to make an informed decision on network protection. The mainstream approach to cyber situation awareness defines three levels that form the awareness [15, 16]: *perception*, *comprehension*, and *prediction*. The perception level serves to obtain data from a monitored environment, the comprehension level aims to understand the information contained in the data, and the goal of the prediction level is to predict the future state of the environment.

IP flow monitoring represents a widely used approach for obtaining network visibility [17]. It processes information from packet headers and aggregates the packets into connection-like records, so-called IP flows [18]. Compared to other methods of network monitoring, e.g., deep packet inspection, the IP flow monitoring enables to process a significantly larger volume of network traffic at higher speeds in exchange for the lower amount of information provided. Due to its properties, IP flow monitoring can be used for gaining cyber situation awareness. It is frequently used for the perception of information from computer networks. Moreover, analysis of the collected IP flows serves well for network comprehension, and even for prediction of the future state of a network, if relevant techniques are applied.

Computer networks are dynamic environments that evolve at a high pace. New technologies are continuously deployed, the paradigms of service and network usage changes in time, and new misuses and attacks are produced rapidly. Such rapid development requires that cyber situation awareness keeps pace with the evolution. The continuous development of computer networks imposed several challenges and open issues that are shared by both cyber situation awareness and IP flow monitoring [16].

Our research work [1] aims to address the current challenges of IP flow monitoring that lead to enhancement of the cyber situation awareness so it can keep up with the development of the computer networks. We propose improvements of IP flow monitoring that enable a better perception of computer networks and we provide the results of several analyses that bring light into network processes and allow for deeper network comprehension.

II. Problem statement and Research Goals

IP flow monitoring was originally designed for network accounting and profiling. The accounting function of IP flow has determined the design of IP flow monitoring infrastructures. IP flows and associated IP flow monitoring infrastructures are designed to provide a holistic view on a network with the focus on regular network reporting. The original design and purpose of IP flow, however, need to be continuously transformed to address technological advances in networking area (faster networks, software defined networks), the shift in protocol usage (rise of encrypted traffic), and new possibilities and paradigms for data processing in general. However, the IP flow definitions, monitoring infrastructure, and related toolset remain with the original purpose, which introduces discrepancies in IP flow applications and other issues in flow processing and analyses.

978-1-7281-4973-8/20/$31.00

We have identified three main open issues of IP flow network monitoring that affects the cyber situation awareness. First, the IP flow monitoring needs to provide sufficient visibility and information for effective cyber defense. Second, the IP flow monitoring is designed for holistic observation of a network from a connection perspective, not for monitoring hosts in a network. Third, requirements for high analysis speed and real-time network monitoring were not included in the design of the traditional IP flow monitoring infrastructure. The open issues are described in detail in the subsections below.

A. Lack of Network Visibility and Comprehension

When we started our research in 2013, the IP flow monitoring provided visibility only into the network and transport layers except for Flexible NetFlow that used NBAR for application recognition. However, the used application recognition only provided the type of an application protocol. No additional application data were available although the data from the application layer represents an information-rich source that enables advanced network analysis and detection of threats targeting applications.

The addition of the application information to the IP flow monitoring increases the volume of the information available on the one hand. On the other hand, it could lead to a performance decline as additional data needs to be retrieved from packets and processed. We believe that the possible performance drop need to be investigated as it could make IP flow monitoring ineffective considering the increasing speed of network traffic. Hence, apart from the increasing the information value of IP flows, we study the trade-off between the increased information value of IP flows and the associated decrease of IP flow monitoring performance.

B. Host Identification in Network Traffic

The connection-based IP flows can be transformend to host-based view. The comprehension of host-based information from the transformed IP flow records is hindered by the fact that a host is represented by an IP address in an IP flow record. A single IP address can represent more hosts in a network. Dynamically addressed networks can assign a different IP address to a host each time it connects to the network. Moreover, network translation devices (NATs) or content delivery networks are represented by a single IP address in IP flow records despite of the fact that they represent more hosts. The fact that there is not only one-to-one host–IP address relation prevents the correct assignment of relevant information from IP flow records to a host. Auxiliary methods for host identification need to be explored so that we do not need to rely soley on IP address as a host's identifier.

Several host's characteristics can be observed in network traffic. If correctly defined, these characteristics could create a host's fingerprint that could provide support for IP address based host identification. The recent rise of the share of encrypted network traffic, however, makes the identification of these characteristics impossible at the application layer. Hence, we study alternative approaches for host identification in encrypted traffic to be able to identify hosts and to obtain a complete view of a network.

C. Delays in IP Flow Monitoring Workflow

As the costs of a service downtime increase, any delay present in the perception, comprehension, or prediction of the computer network that postpones the achievement of the cyber situation awareness is unwelcome. Network IP flow monitoring includes several delays by design, though. A delay occurs during the phase of IP flow creation as the IP flow export is determined by timeouts. Another delay is introduced during IP flow record analysis as IP flow records are analyzed in batches which requires to wait until a batch is complete. These delays can lead to substantial financial losses due to a late response to an attack.

We believe that novel approaches to IP flow metering and analysis need to be researched to obtain a real-time cyber situation awareness and instant response and recovery. We cannot reduce the delay without a significant redesign of the IP flow monitoring workflow. It is essential to keep in mind that the redesign of IP flow monitoring workflow changes the nature of the generated IP flow records. Hence, apart from the redesign of the IP flow monitoring process, also the approach to IP flow analysis needs to be revised to provide real-time cyber situation awareness.

D. Research Goals

The above-described research problems can be summarized into the following main objective of our research:

> *Investigate how IP flow monitoring can be improved to enhance the cyber situation awareness.*

In light of the main objective, we identify the following research goals (RG) that are motivated by the discovered research problems:

- *RG1:* Propose and evaluate IP flow monitoring methods that enhance network perception and comprehension and respond to the emerging trends in the cyber situation awareness and the IP flow monitoring.
- *RG2:* Develop methods for host identification in both unencrypted and encrypted network traffic.
- *RG3:* Provide an option for reducing the delays in the network IP flow monitoring workflow leading to the real-time cyber situation awareness.

The main results of our investigation of these research goals are presented in the sections below.

III. RG1: Enhanced IP Flow Visibility

To enhance the IP flow visibility into a network, we propose the following innovations to the IP flow measurement that provide information from the application layer of the network traffic. First, we investigate IPv6 transition mechanisms and their measurement. Next, we focus on obtaining of information from HTTP protocol. To address the rising share of the encrypted traffic, we propose a technique to obtain information from the HTTPS protocol as well. Last, but not least, we

investigate the trade-off between the increased information value of the IP flows achieved by adding the application layer information and the performance decrease caused by parsing additional information from network.

A. Network Traffic Tunneling

To perform a thorough inspection of tunneled traffic of the IPv6 transition mechanisms, we need to decapsulate packet headers of inner packets. In [2], we extend the approach described in [19] and modify it to extract more detailed information from tunneled data. Resulting IP flow records provide us with information about the encapsulated source and destination addresses, ports and transport protocol, which is a common five-tuple used to distinguish individual flows. We respect this principle and thus we separated the flows encapsulated in the same tunnel based on the value of these elements. Apart from these key elements the framework gives information about *Time to Live* (TTL), *encapsulated HOP limit*, *TCP flags* and *ICMPv6 type* and *code*, when present. Moreover, additional information about tunnel type is provided, including Teredo header and trailer types when present.

Having gained the visibility into IPv6 transition tunnels, we conducted several experiments to uncover the characteristics of the network traffic related to these tunnels. We measured frequency of TTL and HOP values of both encapsulated and encapsulating traffic, location of IPv4 and IPv6 endpoints, distributions of duration and sizes of the observed IP flows, and location of the tunnel endpoints. The Figure 1 shows that the encapsulated traffic origins mainly on the machines with TTL 128, expectedly machines with OS Windows.

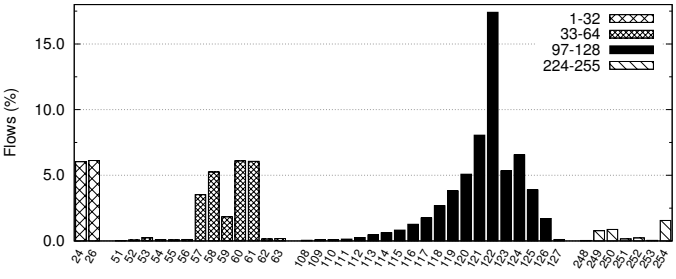

Fig. 1. TTL value distribution of IPv4 traffic that contains IPv6 payload.

B. HTTP Protocol

The HTTP protocol has a number of properties that can be monitored and exported together with IP flow data. The most commonly monitored ones that are present in almost every HTTP request or response header and that we selected for our parsers are: *HTTP method, status code, host, request URI, content type, user agent and referer*.

In [3], we describe our implementation and evaluation of three different parsing algorithms. The first algorithm (*strcmp* approach) loops the HTTP header line by line and searches each line for given fields. It uses standard glibc string functions like *memchr*, *memmem* and *strncmp*. The second algorithm (*pcre* approach) uses several regular expressions taken from

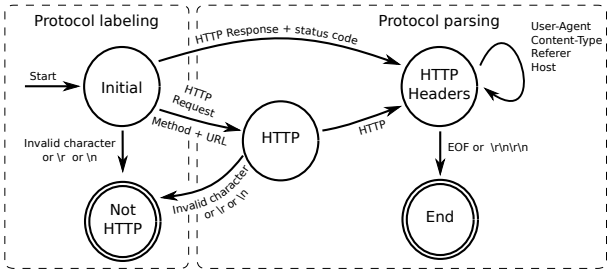

Fig. 2. *flex* algorithm schema.

YAF to search the packet for specific patterns indicating HTTP header fields. We designed the third algorithm (*flex* approach) to handle each packet as a long string. It uses finite automaton to find required HTTP fields, and the Flex lexer is used to process the packets. The automaton design is shown in Figure 2.

C. HTTPS Protocol

Having provided visibility to the HTTP network traffic, we focused on the possibilities of obtaining additional information from encrypted traffic, namely the User-Agents (UA) in the HTTPS traffic [4]. To identify UA clients, we take an approach of creating a dictionary containing pairs of SSL/TLS handshake elements and UAs. We decided to use only a cipher suite list from the *ClientHello* message to build up a dictionary. Cipher suite lists (CS) are the most varied elements of the SSL/TLS handshake, and we supposed that they are sufficient for identifying clients. Other elements of the handshake only have a few different values.

Our approach is based on the extraction of CSs and UAs from SSL/TLS handshake and HTTP traffic respectively, and the correlation of HTTP and HTTPS connections from a single client. We assumed that web clients commonly communicate via both HTTP and HTTPS protocols. We searched for HTTP and HTTPS connections with the same source IP address. We selected a cipher suite list from the HTTPS connections and paired it to the User-Agent from the HTTP connection which was the closest in time to create dictionary of CS-UA relations.

Based on the dictionary, we can then assign the UA to a host communicating over HTTPS using the observed CS list,. The thesis provides several experiments and evaluations of the created dictionary, including the relation of the dictionary size to the covered portion of network traffic, and cardinality of the CS-to-UA relation.

D. Performance-Information Trade-off Evaluation

In the thesis, we provide a performance comparison of the other approaches with our approach and with application-less IP flow probes to investigate the impact of introducing additional information value to the IP flow on the performance of the IP flow monitoring [3]. The result of the throughput comparison is displayed in Figure 3. We recognize an over 50 % decreased performance when application-level information is parsed, and the 300 % better throughput of the

optimized *strcmp* parsing algorithm compared to the others. We further show that the addition of an extra HTTP header information for parsing, once we are already parsing the HTTP protocol, does not significantly decrease the performance of the IP flow probe.

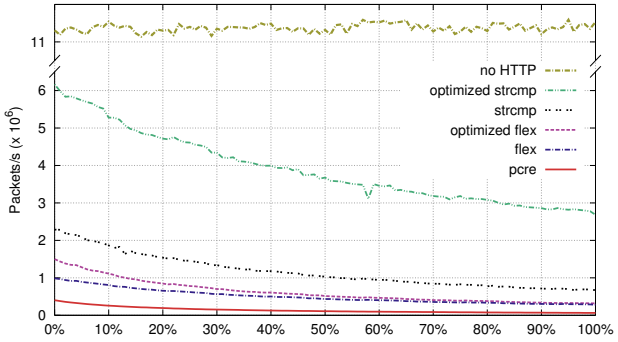

Fig. 3. Parser performance comparison with respect to HTTP proportion (0 % - no HTTP, 100 % - only HTTP headers) in the traffic - full packets 1500 B.

IV. RG2: Host Identification

This section presents our contributions to the host identification in network traffic. As mentioned before, due to NAT devices, dynamically addressed networks, auxiliary host identifiers apart from the IP address are necessary. We focus on host's operating system and type identifiers and investigate the possibilities their identification both in static, dynamic network, and in encrypted network traffic.

A. Static Networks

We compared the uniqueness of the auxiliary identifiers in static and dynamic networks in [5, 6]. During the two hours measurement period assigned for this experiment, we observed 10.221M flows from 12 897 hosts in the campus network. 33.5 % (3.425M) of all monitored flows contained information needed for OS detection which represented 70.33 % (9 072) of all hosts. We observed that in some cases more than one OS was detected for one IP address. The cause of this behavior could be dynamic addressing in networks. Therefore we removed all dynamically addressed subnets from evaluation.

The results show that the portion of the IP addresses with more than one detected OS has decreased after the removal of dynamically addressed networks (see Table I). However, still 4 % of IP shows characteristics of two or more OS even after dynamically address subnets removal. This fact can be explained by the presence of more devices with different OS using the same IP address. This implies the presence of NAT devices. Therefore, the OS detection can also be used as NAT detection assuming that only a static addressed network is monitored.

B. Dynamic Networks

The previous experiment showed the negative impact of dynamic address allocation on the uniqueness of OS. Therefore, we further investigate the possibilities of OS identification

TABLE I
Number of unique OS detected at one IP.

# of OS	# of IP in A	% of all A	# of IP in B	% of all B
1	7898	87.059	3996	95.989
2	1071	11.806	159	3.819
3	80	0.882	7	0.168
> 3	23	0.253	1	0.024
Total	9072	100	4163	100

A - whole network, **B** - dynamically addressed subnets removed

from IP flow records in a dynamically addressed network in [7]. We conduct an experiment to demonstrate the coverage of IP flow based OS detection methods, and evaluate them concerning accuracy, precision, and recall measures. The dataset used in this experiment covers data from all subnets of the wireless network of our university, including buildings of multiple faculties and dormitories. The logs from DHCP servers serves as the ground truth.

We compared the following approaches to the OS identification: TCP/IP parameters (e.g., TTL, SYN packet size, TCP window size) and User-Agent. We further introduced specific domain method for OS identification. It distinguish the OS based on the connections to the well known domains related with a specific OSs, e.g., the OS update servers. The combination of all methods based on the designed weighting system was evaluated as well. The results of the method comparison is presented in Table II with User-Agent showing the best performance.

TABLE II
Micro averaging for multi-class classifier performance measures.

Method	Accuracy	Precision	Recall	F-score
User-Agent	0.9189	0.9812	0.6063	0.7495
TCP/IP parameters	0.8088	0.5249	0.4643	0.4927
Specific domains	0.8402	0.6286	0.4907	0.5512
Combination	0.8582	0.6587	0.6041	0.6302

C. Encrypted Traffic

Using the approach described in the previous section, we are able to identify User-Agent even in the encrypted network traffic. The UA is a rich information source on the devices characteristics. From UA, it is possible to determine the OS of a host, the type of the device, and browser used as well.

Using the dictionary of the CS and UA, we analysed network traffic at university campus to demonstrate the ability to describe the characteristics of the hosts even from encrypted network traffic in [8]. The results show, that the most frequent combination of host type (host/desktop/mobile) and application used are "desktop:browser" with 34.3% followed by the "mobile:browser" reaching the 14.7%. Among other characteristics identified belongs "desktop:command line", or "mobile:application" pairs.

V. RG3: Towards Real-time Cyber Situation Awareness

The delays present in the IP flow monitoring workflow increase time needed to achieve cyber situation awareness. In this section, we show, how can IP flows be analysed in real time. We present an architecture for real-time IP flow monitoring and discuss benefits and pitfalls of the real-time analysis. Last, we outline an approach to real-time cyber situation awareness.

A. Real-time IP Flow Analysis

We present a transformation of the current IP flow monitoring and analysis into the stream-based paradigm in [9]. In this approach, the IP flows are processed and analyzed in data streams immediately after an IP flow is observed. The analysis of IP flows in data streams reduces the volume of data that needs to be stored because data is kept in primary memory for the time necessary for processing and only results are stored in the secondary memory. This feature represents the greatest advantage of the stream-based concept. It allows us to perform the immediate data analysis, which makes the real-time attack detection possible.

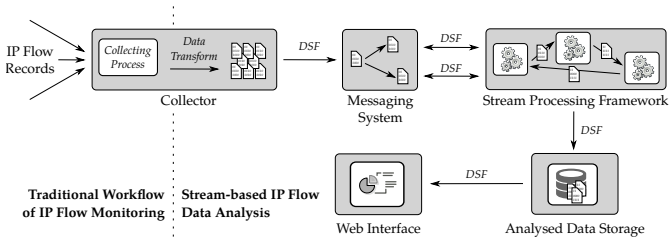

Fig. 4. The architecture of a stream-based IP flow analysis framework.

A general architecture design for stream-based IP flow analysis framework is depicted in Figure 4. The core of the framework is a system for distributed data stream processing, e.g., *Spark*, that is able to process more than 500 000 flows per second [20]. This system is fed with data by a messaging system, e.g., *Kafka*, which enables parallel data distribution, creation of different data streams so data can be analyzed in parallel. The messaging system is connected to a traditional IP flow source. We also present a Stream4Flow framework [10] that implements this design and was deployed in campus network to demonstrate on-the-fly processing of IP flows in large network.

The presented stream-based approach to IP flow analysis brings both several benefits compared to the traditional batch-based approach. There are pitfalls as well. Among the identified benefits belong (1) real-time view into a network traffic so a short burst of network traffic otherwise lost in aggregation can be identified, (2) faster query processing as data are analysed in the primary memory, scalability due to distributed nature of the system, (3) MapReduce programming principle that enables to parallelize computations, increases throughput, and allows to compute hosts characteristics in real-time, for example. The pitfalls connected with the stream-based approach are (1) coherent timestamping of the data in the workflow as wrong timestamps of the data may introduce discrepancies in analyses, (2) increased variability of the data as the observation window is smaller and data are more volatile, and (3) no ex-post queries as the data can be analysed only after a query is formulated in the stream-based approach.

B. Toward Real-time Network-wide Cyber Situation Awareness

Having introduced the architecture for real-time IP flow monitoring, we combine the traditional batch-based approach to IP flow monitoring with the real-time approach to achieve a complex solution that provides real-time network-wide cyber situation awareness in [11]. To combine these two approaches we take advantage of the concept called *lambda architecture.*

The lambda architecture consists of three layers, the batch layer, the speed layer, and the serving layer. The batch layer serves as long term storage and computes batch views on the data. It represents the traditional approach to network monitoring. The speed layer aims to provide real-time views on the data, while the batch view is computed, which is represented by real-time IP flow monitoring. The serving layer combines the real-time views and the batch views together so a user can get query results from both views.

Complex cyber situation awareness demands additional data sources except IP flows – computer logs, network traffic dumps, LDAP details, for example. To be able to process these data in one system, a normalization component needs to be introduced. The normalization transforms collected data from various data sources into a general, interpretable data format. Using this format, information from the different data sources can be analyzed in one system, correlated, and they can provide a complex view on the network. The overview of the framework is presented in Figure 5. More in-depth details on the requirements on the framework for complex cyber situation awareness and the discussion of the proposed framework are provided in [1].

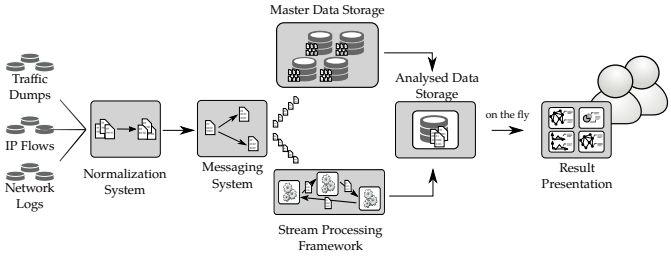

Fig. 5. Framework for perception and comprehension of a network.

VI. Conclusions and Future Work

This paper presents the results of the research [1] with the aim to *investigate, how IP flow monitoring can be improved to enhance the cyber situation awareness*. Our research contributed to the following areas of IP flow monitoring, that improve the overall situation awareness: enhanced network

visibility, host identification in network traffic, and real-time cyber situation awareness based on the real-time IP flow monitoring.

Specifically, we present significant improvements in monitoring HTTP and HTTPS network traffic, including evaluation of the impact on the performance of the IP flow monitoring infrastructure. From the host identification research, we focused on the identification of the auxiliary methods that improve the current approaches. We explored methods for OS identification and validated novel approaches in various types of networks. Last, we investigated how can be a network monitored in real-time. We designed a novel workflow for real-time IP flow monitoring, described the approaches and pitfalls of IP flow analysis in real-time, and generalized this approach to adoption for cyber situation awareness.

Although we significantly contributed to several open issues of cyber situation awareness, there are still remaining opportunities for challenging and interesting future research. Among others, we highlight the following areas worth further investigation: (1) prediction of attacks and attackers intention, (2) correlation of information from various sources, IP flows and logs, for example, (3) stream-based data mining approaches, that enable obtaining complex information in real-time, and (4) host trustworthiness estimation that would provide an operator with a simple measure to asses each host in a network and allow identification of hosts worth close attention.

Publications

[1] Tomas Jirsik. "Cyber Situation Awareness via IP Flow Monitoring". Doctoral thesis, Dissertation. Masaryk University, Faculty of Informatics, Brno, 2019. URL: https://is.muni.cz/th/ejynv/.

[2] Martin Elich et al. "An Investigation into Teredo and 6to4 Transition Mechanisms: Traffic Analysis". In: *Proceedings of Conference on Local Computer Networks, LCN*. 2013, pp. 1018–1024.

[3] Petr Velan, Tomas Jirsik, and Pavel Celeda. "Design and Evaluation of HTTP Protocol Parsers for IPFIX Measurement". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 8115. Springer Berlin Heidelberg, 2013, pp. 136–147. ISBN: 978-3-642-40552-5.

[4] Martin Husak et al. "Network-based HTTPS Client Identification using SSL/TLS Fingerprinting". In: *Proceedings of 10th International Conference on Availability, Reliability and Security, ARES 2015*. 2015, pp. 389–396. ISBN: 9781467365901.

[5] Tomas Jirsik and Pavel Celeda. "Identifying Operating System Using Flow-Based Traffic Fingerprinting". In: *Advances in Communication Networking*. Cham: Springer International Publishing, 2014, pp. 70–73. ISBN: 978-3-319-13488-8.

[6] Tomas Jirsik, Milan Cermak, and Pavel Celeda. "On Information Value of Top N Statistics". In: *2016 6th International Conference on IT Convergence and Security, ICITCS 2016*. 2016, pp. 1–5. ISBN: 9781509037643.

[7] Martin Lastovicka et al. "Passive OS Fingerprinting Methods in the Jungle of Wireless Networks". In: *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*. Taipei, Taiwan: IEEE, April 2018, pp. 1–9. ISBN: 978-1-5386-3416-5.

[8] Martin Husak et al. "HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting". In: *Eurasip Journal on Information Security* 2016.1 (2016). ISSN: 2510523X. DOI: 10.1186/s13635-016-0030-7.

[9] Tomas Jirsik et al. "Toward Stream-based IP Flow Analysis". In: *IEEE Communications Magazine* 55.7 (2017), pp. 70–76. ISSN: 01636804. DOI: 10.1109/MCOM.2017.1600972.

[10] Tomas Jirsik. "Stream4Flow: Real-time IP Flow Host Monitoring using Apache Spark". In: *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*. Taipei, Taiwan: IEEE, 2018, p. 2.

[11] Tomas Jirsik and Pavel Celeda. "Toward Real-time Network-wide Cyber Situational Awareness". In: *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*. Taipei, Taiwan: IEEE, 2018, p. 7.

[12] Milan Cermak et al. "Towards Provable Network Traffic Measurement and Analysis via Semi-Labeled Trace Datasets". In: *2018 Network Traffic Measurement and Analysis Conference (TMA)*. Vienna, Austria, 2018: IEEE, June 2018, pp. 1–8. ISBN: 978-3-903176-09-6.

[13] Petr Velan et al. "Network Traffic Characterisation using Flow-based Statistics". In: *Proceedings of the NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*. 2016, pp. 907–912. ISBN: 9781509002238.

[14] Milan Cermak, Tomas Jirsik, and Martin Lastovicka. "Real-time Analysis of NetFlow Data for Generating Network Traffic Statistics using Apache Spark". In: *Proceedings of the NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*. 2016, pp. 1019–1020. ISBN: 9781509002238.

Acknowledgment

This research was supported by ERDF "CyberSecurity, CyberCrime and Critical Information Infrastructures Center of Excellence" (No. CZ.02.1.01/0.0/0.0/16_019/0000822).

References

[15] M. R. Endsley. "Toward a Theory of Situation Awareness in Dynamic Systems". In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37.1 (1995), pp. 32–64. ISSN: 0018-7208. DOI: 10.1518/001872095779049543.

[16] Alexander Kott, Cliff Wang, and Robert F. Erbacher. *Cyber Defense and Situational Awareness*. Vol. 62. Springer International Publishing, 2015, p. 329. ISBN: 978-3-319-11390-6. DOI: 10.1007/978-3-319-11391-3. URL: http://link.springer.com/10.1007/978-3-319-11391-3.

[17] Rick Hofstede et al. "Flow monitoring explained: From packet capture to data analysis with NetFlow and IPFIX". In: *IEEE Communications Surveys and Tutorials* 16.4 (2014), pp. 2037–2064. ISSN: 1553877X. DOI: 10.1109/COMST.2014.2321898.

[18] B Claise, B Trammell, and P Aitken. *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*. RFC 7011 (INTERNET STANDARD). September 2013. URL: http://www.ietf.org/rfc/rfc7011.txt.

[19] Martin Elich, Matej Gregr, and Pavel Celeda. "Monitoring of tunneled IPv6 traffic using packet decapsulation and IPFIXx (short paper)". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 6613 LNCS. TMA'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 64–71. ISBN: 9783642203046.

[20] Milan Cermak et al. "A performance benchmark for NetFlow data analysis on distributed stream processing systems". In: *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*. IEEE, April 2016, pp. 919–924. ISBN: 978-1-5090-0223-8.