International Journal of Advances in Computer Networks and Its Security– IJCNS
Volume 5: Issue 2 [ISSN : 2250-3757]
Publication Date : 30 October, 2015

Network Monitoring Approaches: An Overview

Jakub Svoboda, Ibrahim Ghafir, Vaclav Prenosil

***Abstract*—Network monitoring and measurement have become more and more important in a modern complicated network. In the past, administrators might only monitor a few network devices or less than a hundred computers. The network bandwidth may be just 10 or 100 Mbps; however, now administrators have to deal with not only higher speed wired network (more than 10 Gbps and ATM (Asynchronous Transfer Mode) network) but also wireless networks. Network administrators are constantly striving to maintain smooth operation of their networks. Network monitoring is a set of mechanisms that allows network administrators to know instantaneous state and long-term trends of a complex computer network. This paper provides the readers with an overview of the current network monitoring approaches, their architectures, features and properties. In addition, it presents a comparison between those approaches.**

***Keywords*—Network monitoring, traffic duplication, packet capture, deep packet inspection, flow observation.**

I. Introduction

Computer networks are connecting millions of computers and computer users throughout the world. The network has become an infrastructure for many applications that affect our daily lives. It is very important that the computer network needs to be managed properly. Management of networking requires monitoring. Network monitoring is a set of mechanisms that allows network administrators to know instantaneous state and long-term trends of a complex computer network.

Network monitoring and measurement have become more and more important in a modern complicated network. In the past, administrators might only monitor a few network devices or less than a hundred computers. The network bandwidth may be just 10 or 100 Mbps; however, now administrators have to deal with not only higher speed wired network (more than 10 Gbps and ATM (Asynchronous Transfer Mode) network) but also wireless networks. They need more sophisticated network traffic monitoring and analysis tools in order to maintain the network system stability and availability such as to fix network problems on time or to avoid network failure, to ensure the network security strength, and to make good decisions for network planning.

Network Monitoring involves multiple methods which are deployed on purpose to maintain the security and integrity of an internal network. The internal network is also known as a Local Area Network (LAN) and monitoring encompasses hardware, software, viruses, spyware, vulnerabilities such as backdoors and security holes, and other aspects that can compromise the integrity of a network.

Network monitoring is a difficult and demanding task that is a vital part of a network administrators job. Network administrators are constantly striving to maintain smooth operation of their networks. If a network were to be down even for a small period of time, productivity within a company would decline, and in the case of public service departments the ability to provide essential services would be compromised. In order to be proactive rather than reactive, administrators need to monitor traffic movement and performance throughout the network and verify that security breaches do not occur within the network.

When a network failure occurs, monitoring agents have to detect, isolate, and correct malfunctions in the network and possibly recover the failure. Commonly, the agents should warn the administrators to fix the problems within a minute. With the stable network, the administrators' jobs remain to monitor constantly if there is a threat from either inside or outside network. Moreover, they have to regularly check the network performance if the network devices are overloaded. Before a failure due to the overload, information about network usage can be used to make a network plan for short-term and long-term future improvement.

This paper provides the readers with an overview of the current network monitoring approaches, their architectures, features and properties. In addition, it presents a comparison between those approaches.

The remainder of this paper is organized as follows. Section 2 classifies the current network monitoring approaches. A comparison between the presented approaches is provided in Section 3. Section 4 concludes the paper.

II. Network Monitoring Approaches

Network monitoring can be either active or passive. Passive network monitoring reads data from the line, without affecting the traffic. Active network monitoring adds option to modify the data on the line [1].

Passive network monitoring exists in several forms. Simple monitoring may be easy for manual assessment as the amount of data monitored and produced is small. Monitoring of all sorts of details about the network and its traffic bears a similar hurdle; information about faults and attackers are gathered, but there is so much information that it gets lost in the sea. Also,

Jakub Svoboda [1], Ibrahim Ghafir [2], Vaclav Prenosil [2]
[1] Institute of Computer Science, Masaryk University
[2] Faculty of Informatics, Masaryk University
Brno, Czech Republic

the more data captured, the more technologically demanding it is to save and handle the data. Therefore, various ways of doing network monitoring compete with each other, as each has different tradeoffs, being targeted for different purposes, environments and users. Figure 1 shows the general architecture of network monitoring.

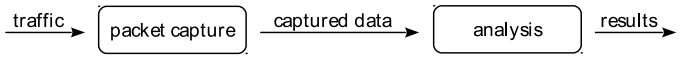

Figure. 1. General architecture of network monitoring.

The process of network monitoring consists of two major steps, traffic duplication and traffic analysis.

A. *Traffic Duplication*

All types of network traffic monitoring have one common property; the traffic from the line is duplicated so that the copy can be analyzed. The duplication can take place in one of two modes; inline or mirroring [2]. A traffic duplication device in inline mode is placed in link. In mirroring mode, the duplication facility is already a built-in feature of a router or switch. There are several ways of traffic mirroring; port mirroring, TAP and a TAP-like setup using bypass NICs. The following subsections describe each way.

1) Port Mirroring

Port mirroring is a functionality usually available in enterprise-oriented network switches and routers [3]. The traffic passing through selected ports of the switch or router is mirrored to another selected port. The port used for output of the duplicated traffic is usually called mirror port or SPAN port (Switched Port Analyzer). Figure 2 shows the principle of port mirroring. Both directions of the monitored link are transmitted in one direction over the mirror port.

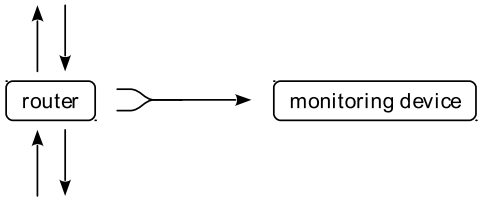

Figure. 2. Principle of port mirroring.

There are two downsides of mirror ports. First, if the sum throughput of the traffic is larger than the mirror port can transmit, the mirror port becomes congested and drops packets. A full-duplex traffic is transmitted in one direction over the mirror port. That is up to twice the bandwidth of a single port for two ports serviced by the switch, and even more if more than two ports are serviced [4]. Second, most switches do not have enough computational power to handle both switching and mirroring. The switch's primary function is prioritized and the mirroring may not work properly during periods of peak traffic.

2) Test Access Port

Test Access Port (TAP) is a packet capture device positioned in inline mode since the observed line is split. A TAP device is connected between the split parts of the line and the traffic is duplicated. Single TAPs duplicate the traffic to a single output, consisting of two physical ports for both downlink and uplink of the full-duplex link. Regeneration TAPs duplicate the traffic into multiple outputs. Aggregation TAPs merge both channels into one output port. There are three types of TAPs; copper, fiber and virtual. Figure 3 shows the traffic mirroring using Test Access Port. Both directions of the monitored link are transmitted separately.

Passive copper TAPs connect directly to the line. Since passive TAPs are not powered, a power outage cannot introduce a fault on the line. A disadvantage to passive copper TAPs is that only 10-Mbps and 100-Mbps connections are possible to tap this way. The passive connection distorts the signal in such a way that it is not possible to tap a gigabit Ethernet passively [5]. A patent by NetOptic presents a method that uses an active gigabit TAP equipped with capacitors to maintain the connection while the built-in bypass relays are switching [6].

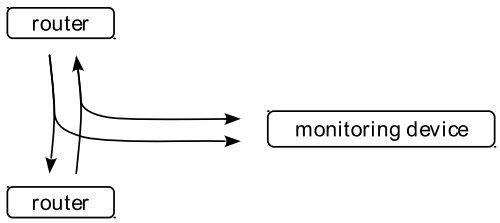

Figure. 3. Traffic mirroring using Test Access Port.

Active copper TAPs function in a way resembling the approach in the preceding subsection. The signal going through the TAP is actively retransmitted and duplicated and no signal distortions are introduced, short of the negligible delay caused by the electronic circuitry. Downside of this approach is that power failure of the TAP causes a failover relay to switch, introducing a several hundred microseconds long delay [7].

Passive optical TAPs divert a percentage of the original signal to the mirrored output. The fact that no power failure of the TAP can occur is an advantage. A disadvantage lies in the fact that the signal in the line is weakened by the TAP [8].

Regeneration optical TAPs divert a very small fraction of the original signal to the mirrored output and amplify it to the full strength. A power failure just disables the mirrored output and introduces no faults on the line.

3) TAP-like Setup Using a Bypass NIC

Setup using a network interface card (NIC) integrates traffic mirroring with traffic analysis. As it is shown in Figure 4, the observed line is split. Both ends in the split are connected to two NIC interfaces. The NIC is installed in a computer. The interfaces are configured in software as a network bridge. Acting as a bridge allows the split line to still

function properly. Having the traffic pass through the computer allows traffic observation. This setup is positioned in inline mode, similarly to a TAP.

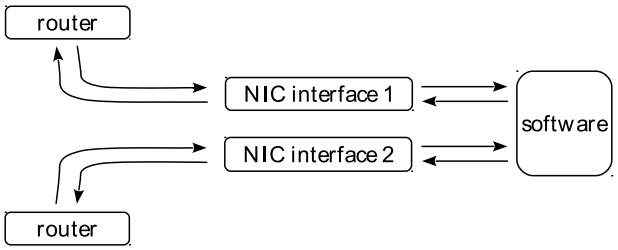

Figure. 4. Inline mirroring using two interfaces of a NIC.

Mirroring using NIC is possible using consumer-grade NICs. This introduces a point of failure. Once the software or hardware fails, the line is not connected anymore.

Specialized, so-called bypass NICs exist, as in Figure 5. Bypass NICs have the ability to bypass the two network interfaces whenever a failure occurs; e.g., a software crash or power loss [9].

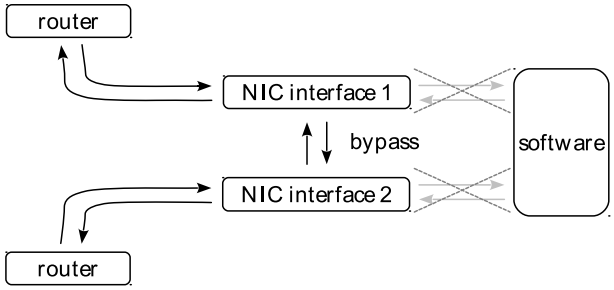

Figure. 5. Bypass NIC in bypass mode bridges the connection in hardware so as not to break the connection.

A disadvantage is that the computer is locked in the particular location and cannot be moved without interrupting the connection.

B. Packet Capture

Packet capture has three meanings in no particular order. First, it is an interactive approach to network monitoring. Second, packet capture is a packet trace file. Third, it is the act of capturing packets from network link. The capture can be saved to a file or read directly by a network traffic analyzer in real time [10].

1) Packet Capture as a Packet Duplication Method

Network traffic is captured from an observation point. It is not necessary that the capture is temporally and spatially dependent on subsequent analysis since the captured data can be saved to a file. This can be a temporary file as a part of the whole network monitoring process or it can be saved to a file for later use explicitly. The captured data is the same as it was on the line. The process of making a packet capture can be both manual and automatic [11].

2) Packet Capture as a Network Monitoring Approach

The packet capture network monitoring approach consists of two basic steps; first, creating the packet capture file, and second, performing network traffic analysis on the captured file. Packet capture as a network monitoring approach can be both manual and automated. The automated approach is used for malware behavior recording and analysis [11]. Additional manual analysis of selected packet captures from such an automated system may also be possible [12].

The Layer 3 of the OSI model is usually used so that the traffic is seen as a series of IP packets. The captured traffic in this representation can then be viewed, searched in, or filtered [13]. It is also possible to filter the packets before capturing the packet trace [14].

Both graphical user interface (GUI) and command line interface (CLI) are used. In some setups, automation through scripting of the actions is possible. This is merely intended as a help for the human user and not designed to implement a complicated automated system. An intrusion detection system (IDS) may be considered to be a complicated automated system. In the context of packet capture analysis, even an IDS-like scripted functionality is still intended for individual interactive analysis [15].

The accessibility of full network data for free viewing and searching is unsurpassed among all the architectural approaches mentioned in this paper. The interactivity and access to any part of the traffic data can be an immense advantage. The user can search for highly specific artifacts without having to program anything and without being constrained by more automated software. It is useful for dealing with new traffic patterns, such as new malware or unknown communication protocols [12]. The interactivity is, however, a disadvantage when the pattern is already known, and the work is repetitive and automatable. The approach does not scale and it becomes burdensome to search through large amounts of data.

PCAP [16] is a commonly used file format for storage of the captured traffic. Tshark [17] and tcpdump [18] are examples of software operated through CLI. Wireshark [19] is an example of software with GUI.

C. Deep Packet Inspection

Automation is a characteristic feature of deep packet inspection (DPI), especially in comparison to manual packet capture and analysis described above. Deep packet inspection is a technique of seeing the payload of IP packets. It is, however, also used to denote those architectural approaches to network traffic monitoring that use DPI in an automated fashion; DPI is incorporated into inherently automated systems.

Traffic capture and further analysis can be either separate processes in both time and space or they can be integrated in one process pipeline, as it is shown in Figure 6. The packet

capture approach can serve as a source of a PCAP file for further DPI-based analysis [20].

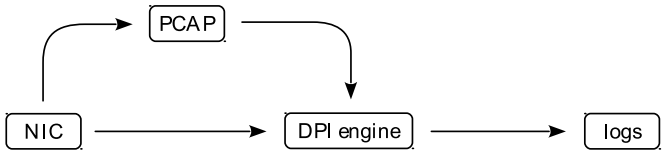

Figure. 6. DPI-based approach.

There are two major types of DPI-based analysis; pattern matching and event-based analysis. Both are used in various IDS/IPS (Intrusion Detection Systems / Intrusion Prevention Systems).

1) Pattern Matching

Pattern matching is a DPI method that involves searching through full network data for known sequences of bytes or for regular expression matches [21]. The principle of operation is shown in Figure 7. The search can be limited to specific parts of packets or to specific packets.

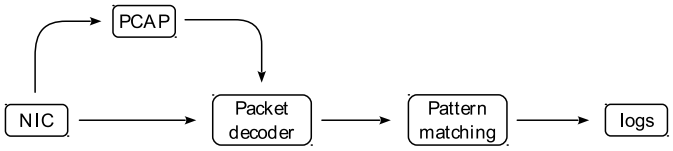

Figure. 7. DPI – Pattern matching method.

The relative simplicity is an advantage of this approach, which is why it is a popular type of DPI. Describing the sought data by sequences of bytes or by regular expressions is often straightforward.

This strength, however, becomes a problem when we want to search for patterns that are not possible or feasible to describe using regular expressions. If the data has to be decoded before further pattern matching and the decoding functionality is not already built in the network security monitor, it is usually impossible to craft a regular expression that also does the decoding. Compression may serve as an example of such decoding necessary before the pattern matching.

Complicated decision logic is also infeasible to do using just regular expressions. Example task; alert on expired SSL certificates on HTTPS connections coming from a specified list of IP addresses and from nowhere else. Translating detection of SSL certificate data to regular expressions might be impossible. The check whether a specific certificate belongs to a list might result in a complicated regular expression. Even if the first problem is ameliorated by an SSL decoder, the second one still stands. One step further, if the list of cues is dynamically changing at runtime, conversion of the algorithm to a regular expression becomes impossible at all. An example for that might be a threshold-based detection, e.g., alerting on hosts receiving more than 10 DNS errors per hour. Today's network traffic monitors, employing the pattern matching DPI method, usually decode the most used protocols [22].

The pattern matching approach is slow, compared to the flow observation approach explained in the following section. A concrete pattern matching implementation for 10 Gbps requires hardware acceleration using FPGA [23]. In contrast, a concrete flow observation implementation with no hardware acceleration handles 40 Gbps [24]. Compared to the event based approach in the next subsection, the pattern matching approach is also rather simplistic.

There are numerous algorithms for pattern matching. Pattern matching algorithms in the context of network monitoring are described by J. Kelly [25]. Besides algorithms, there are software packages for pattern matching, ready to be built into other software; Flex and MultiFast for instance. Use of Flex and MultiFast in the context of network traffic analysis is researched by T. Šima [26].

Snort [27] and Suricata [28] are software implementations of pattern matching DPI. ngrep [29] is a command line utility for pattern matching in captured packet traces.

2) Event-based Analysis

The previous subsection describes cases where pattern matching is clearly an insufficient technique. Its inability to perform decoding or multiple steps of decision making is addressed in the architectural approach of event-based analysis. In the approach of DPI with event-based analysis, as it is shown in Figure 8, packets are processed into events that are in turn processed by scripts [30]. Scripts may implement complex processing algorithms and add new DPI-related functionality.

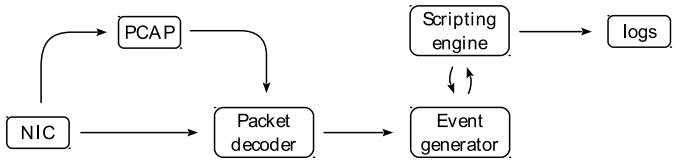

Figure. 8. DPI with event-based analysis.

Such an architectural approach replaces the pattern matching part with algorithms implemented as a computer program. The algorithms can be both stateful and stateless. Stateless algorithms are just an immediate reaction or chain of reactions to specific events. Stateful algorithms can use program variables to remember state between event occurrences. Bro Network Security Monitor [31] is a network monitor with such an architecture.

D. *Flow Observation*

An approach differing from the ones described in preceding sections is flow observation. The contents of the packets are not analyzed beyond information from packet headers. This information is aggregated into flows. RFC 7011

[32] provides the following definition of a flow: "A Flow is defined as a set of packets or frames passing an Observation Point in the network during a certain time interval. All packets belonging to a particular Flow have a set of common properties."

The following five-tuple is used as the common property distinguishing flows from each other: source IP address, destination IP address, source IP port, destination IP port, Layer 4 protocol.

In the architectural approach of flow observation, the network traffic monitor stores information about the observed flows like the identifying 5-tuple, number of transmitted bytes and packets and L4 protocol flags, but it does not analyze nor store the payload.

Because the data itself is not handled, flow observation has several advantages. Since the payload is not analyzed, flow observation is faster than the other approaches on the same hardware. Moreover, not storing the payload results in considerably less stored data than in the case of packet capture mentioned above. Not processing the payload also makes it less of a privacy concern, compared to packet capture or DPI. Flow data may be used to comply with data retention laws [33].

Figure 9 shows the flow observation architecture. Upon observation, packets are sent to a metering process. The metering process identifies flows and counts their statistics. From this point on, the original packets are not processed. The metering process sends the information about flows to an export process after a certain period of time. The metering and export processes usually reside together on a network probe. The export process sends the finalized flow information to a collector for storage and subsequent processing.

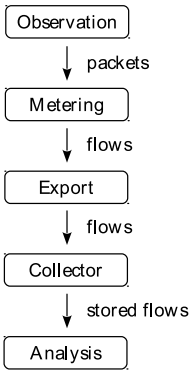

Figure. 9. Flow observation architecture.

There are two notable formats for the transmission of flow information; NetFlow [34] developed by CISCO and IPFIX [35] developed by IETF. Some of the existing software flow exporters are nProbe [36] softflowd [37], YAF [38], and FlowMon [39]; selected flow collectors are nProbe [36], nfdump [40], flowd [41], IPFIXcol [42], and SiLK [43].

III. Comparison

Table I shows a comparison between the current network monitoring approaches and presents the pros and cons of each approach.

TABLE I. COMPARISON BETWEEN NETWORK MONITORING APPROACHES

Monitoring Approach	Pros and Cons	
Port Mirroring	Pros	- Widely available in switches.
	Cons	- Unreliable.
TAP	Pros	- Reliable at high speeds.
	Cons	- Expensive. - Requires disconnection at installation.
Packet capture	Pros	- Unfettered access to full network data during analysis.
	Cons	- Mostly manual analysis. - Doesn't scale to high speeds and large capture size.
Pattern matching DPI	Pros	- Easy detection rule development. - Scales well to high speeds.
	Cons	- Not all data can be described by patterns or regular expressions.
Event-based DPI	Pros	- Scales well to high speeds. - Improvement in expressive power over pattern matching DPI.
	Cons	- Requires more complicated implementation than pattern matching DPI. - Rule development requires the developer to learn considerably more about the DPI implementation than for pattern matching.
Flow observation	Pros	- Privacy – packet payload data not used. - Scales well to high speeds. - The output data is a fraction of the size of the monitored traffic data.
	Cons	- Stores only aggregated metadata about the traffic. - Limited options at analyzing data.

IV. Conclusion

Given the data packet and network traffic flow information, administrators can understand network behavior, such as application and network usage, utilization of network resources, network anomalies and security vulnerabilities. The network traffic is first duplicated and then analyzed. There are two ways of traffic duplication; port mirroring and TAPs. There are three approaches to traffic analysis; packet capture, predominantly automated deep packet inspection, and flow observation. Each approach has its strengths and disadvantages.

This paper provides the readers with an overview of the current network monitoring approaches, their architectures, features and properties. In addition, it presents a comparison between those approaches.

Acknowledgment

This work has been supported by the project CYBER-2 funded by the Ministry of Defence of the Czech Republic under contract No. 1201 4 7110.

References

[1] Cottrell, L.: Passive vs. Active Monitoring [online]. [cit. 2015-04-21]. URL, https://www.slac.stanford.edu/comp/net/wanmon/passive-vs-active.html.

[2] Worrall, A.; Carter, B.;Widley, G.: Network monitor and method. 2008 [cit. 2015-04-21], URL, http://www.google.com/patents/US7411946.

[3] CaptureSetup/Ethernet – TheWiresharkWiki [online]. [cit. 2015-04-21]. URL, http://wiki.wireshark.org/CaptureSetup/Ethernet.

[4] Cisco Systems, Inc.: Catalyst Switched Port Analyzer (SPAN) Configuration Example Cisco [online]. [cit. 2015-04-21]. URL, http://www.cisco.com/c/en/us/support/docs/switches/catalyst-6500-series-switches/10570-41.html.

[5] Leong, P.: Ethernet 10/100/1000 Copper Taps, Passive or Active? [online]. [cit. 2015-04-21]. URL, http://www.lovemytool.com/blog/2007/10/copper-tap.html.

[6] Matityahu, E.; Shaw, R.; Carpio, D.; aj.: Gigabits zero-delay tap and methods thereof. 2011, [cit. 2015-04-21], uS Patent App. 13/034,730. URL, http://www.google.com/patents/US20110211446.

[7] Datacom Systems: Choosing a Network TAP [online]. [cit. 2015-04-21]. URL, http://justnetworktaps.com/article_info.php?articles_id=3.

[8] JDSU Storage Network Test: Understanding Fibre Optic Network Tapping [online]. [cit. 2015-04-21]. URL, http://www.jdsu.com/ProductLiterature/Understanding-Fiber-Optic-Network-Tapping-white-paper-30162800.pdf.

[9] PE210G2BPi9 Dual Port Fiber 10 Gigabit Ethernet PCI Express Bypass Server Adapter Intel® based [online]. [cit. 2015-04-21]. URL, http://www.silicom-usa.com/upload/Downloads/Product_95.pdf.

[10] Shepard, T. J.: TCP Packet Trace Analysis. Technical report, Laboratory for Computer Science, Massachusetts Institute of Technology, 1991.

[11] Baecher, P.; Koetter, M.; Holz, T.; aj.: The Nepenthes Platform: An Efficient Approach to Collect Malware. In Recent Advances in Intrusion Detection, Springer Berlin Heidelberg, 2006, ISBN 978-3-540-39723-6.

[12] Balas, E.; Viecco, C.: Towards a third generation data capture architecture for honeynets. In Information Assurance Workshop, 2005. doi:10.1109/IAW.2005.1495929.

[13] INTECO-CERT: TRAFFIC ANALYSIS WITH WIRESHARK [online]. [cit. 2015-04-21]. URL, http://www.csirtcv.gva.es/sites/all/files/downloads/cert_trafficwireshark.pdf.

[14] McCanne, S.; Jacobson, V.: The BSD Packet Filter: A New Architecture for User level Packet Capture. In Proceedings of the USENIX Winter 1993 Conference, Berkeley, CA, USA: USENIX Association, 1993.

[15] Arcas, G.: WireShnork - A Snort plugin for Wireshark [online]. [cit. 2015-04-21]. URL, http://honeynet.org/node/790.

[16] Development/LibpcapFileFormat -TheWiresharkWiki. [cit. 2015-04-21]. URL, http://wiki.wireshark.org/Development/LibpcapFileFormat.

[17] D.2. tshark: Terminal-basedWireshark [online]. URL, https://www.wireshark.org/docs/wsug_html_chunked/AppToolstshark.html.

[18] TCPDUMP/LIBPCAP public repository [online]. [cit. 2015-04-21]. URL, http://www.tcpdump.org/.

[19] Wireshark - About [online]. [cit. 2015-04-21]. URL, https://www.wireshark.org/about.html.

[20] Kumar, S.; Sehgal, R.; Bhatia, J. S.: Hybrid honeypot framework for malware collection and analysis. In Industrial and Information Systems (ICIIS), 2012 7th IEEE International Conference.

[21] Lin, P.-C.; Lin, Y.-D.; Lee, T.; aj.: Using string matching for deep packet inspection. 2008 [cit. 2015-04-21]. URL, http://speed.cs.nctu.edu.tw/~ydlin/string%20matching.pdf.

[22] Decoder and Preprocessor Rules [online]. [cit. 2015-04-21]. URL, http://manual.snort.org/node18.html.

[23] Franklin, M.; Crowley, P.; Hadimioglu, H.; aj.: Network Processor Design: Issues and Practices. The Morgan Kaufmann Series in Computer Architecture and Design, Elsevier Science, 2005, ISBN 9780080512501.

[24] INVEA-TECH a.s.: FlowMon Probe Models List [online]. [cit. 2015-04-21]. URL, https://www.invea.com/data/flowmon/flowmon_probe_specification_en.pdf.

[25] KELLY, J.: An Examination of Pattern Matching Algorithms for Intrusion Detection Systems. Master's thesis, Ottawa Carleton Institute for Computer Science, Carleton University, Canada, 2006.

[26] ŠIMA, T.: Mereni HTTP a HTTPS provozu pomoci IP toku [online]. Bachelor's thesis, Faculty of Informatics, Masaryk University, Czech Republic, 2014.

[27] Snort. Home Page [online]. [cit. 2015-04-21]. URL, http://www.snort.org/.

[28] Suricata Open Source IDS / IPS / NSM engine [online]. [cit. 2015-04-21]. URL, http://suricata-ids.org/.

[29] Ritter, J.: ngrep - network grep [online]. [cit. 2015-04-21]. URL, http://ngrep.sourceforge.net/.

[30] The Bro Network Security Monitor [online]. [cit. 2015-04-21]. URL, http://www.bro.org/documentation/overview.html.

[31] Paxson, V.. Bro: A System for Detecting Network Intruders in Real-time. Comput. Netw., 1999: s. 2435 − 2463, ISSN 1389-1286, doi:10.1016/S1389-1286(99)00112-7.

[32] Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information Types and attributes [online]. [cit. 2015-04-21]. URL, http://tools.ietf.org/search/rfc7011.

[33] INVEA-TECH a.s.: FlowMon Data Retention [online]. [cit. 2015-04-21]. URL, https://www.invea.com/data/flowmon/flowmon_data_retention_pb_en.pdf.

[34] Claise, B.: Cisco Systems NetFlow Services Export Version 9 [online]. [cit. 2015-04-21]. URL, https://tools.ietf.org/html/rfc3954.

[35] Trammell, B.; Claise, B.: Flow Aggregation for the IP Flow Information Export (IPFIX) Protocol [online]. [cit. 2015-04-21]. URL, https://tools.ietf.org/html/rfc7015.

[36] nProbe [online]. [cit. 2015-04-21]. URL, http://www.ntop.org/products/nprobe/.

[37] softflowd - A software NetFlow probe [online]. [cit. 2015-04-21]. URL, https://code.google.com/p/softflowd/.

[38] Christopher Inacio, B. T.: YAF: Yet Another Flowmeter [online]. [cit. 2015-04-21]. URL, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.368.3172&rep=rep1&type=pdf.

[39] INVEA-TECH: FlowMon − Network monitoring [online]. [cit. 2014-05-21]. URL, http://www.invea.com/en/products-and-services/flowmon>

[40] NFDUMP [online]. [cit. 2015-04-21]. URL, http://nfdump.sourceforge.net/.

[41] flowd - small, fast and secure NetFlow collector [online]. [cit. 2015-04-21]. URL, http://code.google.com/p/flowd/.

[42] IPFIXcol | Liberouter / Cesnet TMC group [online]. [cit. 2015-04-21]. URL, https://www.liberouter.org/technologies/ipfixcol/.

[43] Gates, C.; Collins, M.; Duggan, M.; aj.: More Netflow Tools for Performance and Security. In Proceedings of the 18th USENIX Conference on System Administration, USENIX Association, 2004.