Network probe: Network monitoring and management tool

A. Bohačík, R. Fujdiak, and J. Mišurec

Brno University of Technology, FEEC, Czech Republic

E-mail: Antonin.Bohacik@vut.cz

Abstract—Nowadays, there are many risks associated with computer networks, some of them can be eliminated with network probes. This paper is focused on the developed BUT network probe as a tool representing a hardware protection element of the network. Furthermore, the basics of IDS and IPS systems are described, including their possible applications. The basic concept of the network probe, the description of its basic parts and the created user interface are discussed. The last part is focused on the testing of hardware components that directly affect the proper functioning. The test results showed that the BUT network probe is able to perform network traffic analysis even at its maximum load.

Keywords— IDS, IPS, network analysis, network probe, Suricata

1. INTRODUCTION

With the development of computer technology and Internet technologies, the threat of remote attacks also began to develop [1]. This threat has resulted in the creation of specific organizations dealing with protection against computer attacks, such as anti-virus companies or state security authorities. Today, most attacks are carried out remotely using the Internet [2]. Therefore, it is often difficult to find the real attacker behind the attack. For these reasons, we use IDS (Intrusion Detection System) and IPS (Intrusion Prevention System) systems [2, 3, 4]. As the names suggest, these are systems that are used to detect or block specific network traffic [4]. Currently, there are many solutions to prevent a remote attack, but most solutions, such as the firewall, have a negative effect on the performance of the device itself [1, 2]. On the contrary, there are network devices that provide detection (or blocking) directly during data transmission, but most of these devices are expensive and mostly used only by large companies or organizations [4, 5].

Network probes can be a compromise [6]. These are devices that communicate with network devices (e.g. routers) and inform an authorized user (usually a network administrator) about certain events at the real time. This is the so-called real-time traffic analysis, which can prevent more serious damage to the monitored network [2, 4, 7]. Network probes can also combine various features of other systems, such as recording of transmitted data, traffic filtering, or analysis of specific protocols. Another advantage is the affordability for smaller companies or organizations [6]. Table I compares some existing solutions with comparable performance for analyzing, recording, or filtering traffic, including their pricing.

Device	Throughput	Data recording	Storage capacity	IDS	IPS	Price [€]
BUT network probe	1 Gb/s	✓	64 GB + external	✓	✓*	1 600
TAP Profishark 1G	1 Gb/s	✓	only external	✗	✗	2 800
Cisco FirePOWER 2110 NGFW	2 Gb/s	✓	100 GB	✓	✓	5 200
IPCopper models	1-12 Gb/s	✓	1-48 TB	✓	✓	5 200-23 400
Flowmon	SaaS**	✗	✗	✓	✓	9 000/year

**Not fully implemented.*

***Software as a service – Baud rate depends on the infrastructure and the agreed price.*

Table I: Comparison of some solutions for network analysis

2. NETWORK PROBE

A network probe (see Figure 1) was created at the Department of Telecommunications for teaching and testing purposes. This probe is a tool for detecting cyber and security threats in real time. It can be used to protect the internal network terminal equipment or to analyze network traffic at a given point. The detection of defined attack signatures is used for the analysis of network communication, but the software solution enables the analysis of suspicious activities, so the probe is able to recognize even unknown types of attacks. The probe also includes an external hard disk for storing recorded data using the implemented T-shark module.

A single-board computer called 4x4-4500U from ASrock-Industrial was chosen for the network probe. The device has two Realtek R8111FPV (enp1s0) and Realtek RTL8125BG (enp2s0f0) network cards used as the primary interface for traffic analysis. Among other things, it also contains two USB connectors of type C, which support the connection of powerful external network adapters for possible needs of processing traffic from multiple inputs. Due to the great flexibility of settings, the operating system Ubuntu 20.04 was chosen. Traffic filtering is implemented by using IPtables and its successor NFtables. These tools allow you to create specific rules that determine which data a device should receive, discard, or further process. Signatures are detected using the implemented NIDS system [3, 4, 7].

The internal block diagram of the probe is shown in Figure 2. All data traffic intended for processing is sent to the Suricata module. It analyzes individual messages and compares them with the internal signature database that is a part of the web application. The continuous module stores action logs, which are further operated by the web server.

Figure 1: BUT network probe

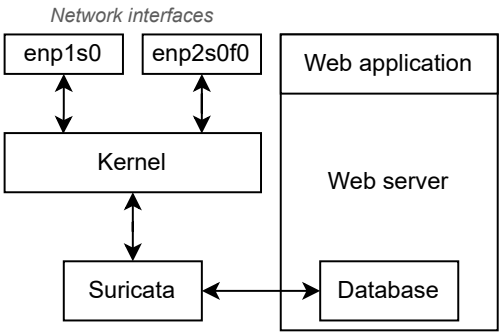

Figure 2: Block diagram of the network probe

The network probe has a multi-platform web interface (see Figure 3), which allows the user to control and configure the device (e.g. add rules, delete logs, start traffic logging, etc.). The basis of the web interface is the open-source web framework Django. A characteristic feature of this framework is the Model-View-Controller architecture. The model represents a database of user data, the view represents a graphical representation of the control elements and the controller is used to control individual elements. The interface consists of six basic parts and contains modules that provide the required functionality. The interface is used to communicate with the Ubuntu OS, create individual processes, or rewrite Suricata configuration files, which is used for network traffic analysis.

In addition to the basic ability of the network probe to process network traffic according to the set rules, the Scappy library was implemented in the probe. It is used for detection and partly analysis of various industrial protocols. Supported protocols that can be detected by the probe include IEC 60870-5-104, Device Language Message Specification (DLMS), protocols from IEC 61850 standard (GOOSE and Sample Value). These protocols were chosen because of their frequent deployment in practice.

3. PERFORMANCE TESTING

The functionality and usability of the probe itself depends on the individual network cards. For these purposes, the RFC 1242 and RFC 2544 standards were used, which define the methodology for testing

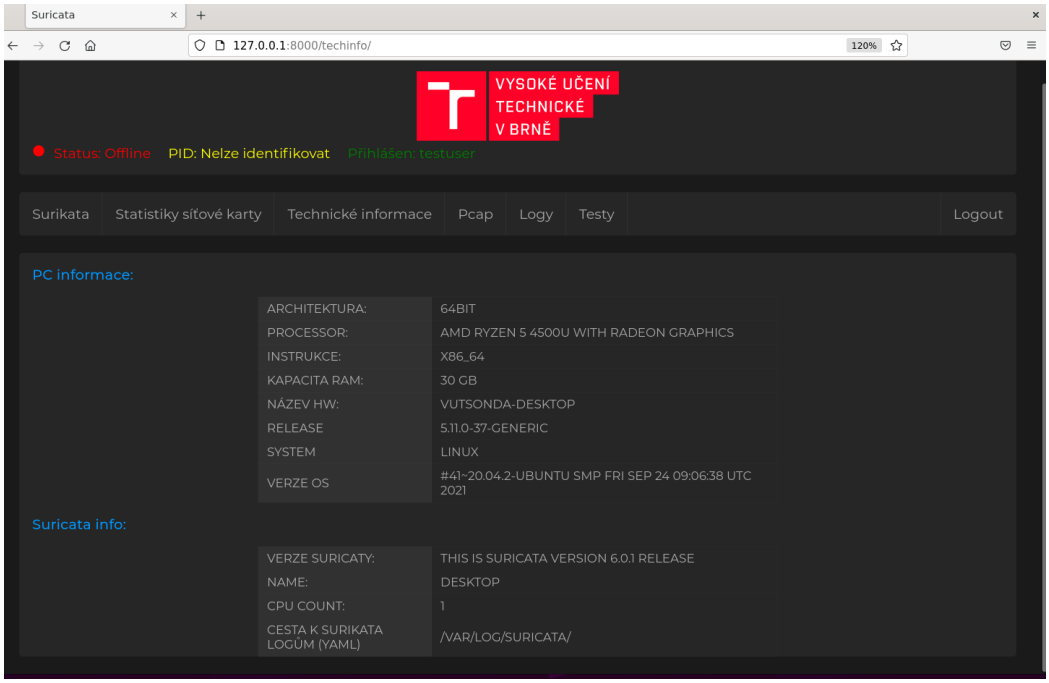

Figure 3: Web application – Information page

network parameters (e.g. throughput, latency, etc.). The aim of the tests was primarily to verify hardware limits of the device, which include: maximum processing speed of input/output data, load of operational memory or processor.

The internal BUT payload generator (PLG) was used to test the network probe. This generator contains an Intel Xeon E5-2650 v4 @2.2 GHz CPU with 128 GB of RAM and an NVIDIA GeForce GT 730 VGA graphics card. It also includes 5 network interfaces, 2 x 1 Gb/s (on the motherboard), 2 x 10Gb/s (Intel X540-AT2) and 2 x 10 Gb/s (Intel 82599ES SFI/SFP+). The whole device runs on CentOS7 operating system. Due to the nature of testing, this generator was set for all tests to generate TCP packets at a rate of 200 000 packets/s (so-called TCP Flood). The size of one packet was set to 1 460 B. Such a setting generates a load of approximately 2.336 Gb/s for one network interface. Two connection schemas were used for the testing. The schema shown in Figure 4 was used to test the hardware components. The schema in Figure 5 shows the so-called bridge mode, which was used to test the probe's ability to detect the set Suricata rules. This mode software combines *enp1s0* and *enp2s0f0* interfaces into one virtual *br0*.

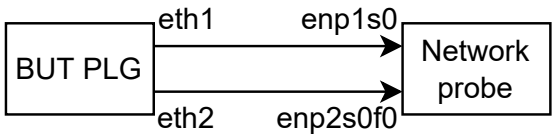

Figure 4: Performance testing schema

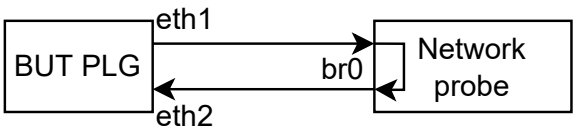

Figure 5: Bridge mode schema

3.1. Hardware components test

The first set of tests was focused on the performance of the individual components of the network probe. These were mainly the CPU (Central Processing Unit), RAM (Random Access Memory) and network interfaces. These components directly affect the usability of the network probe. Testing was performed on the schema shown in Figure 4. In order not to affect the results, all unnecessary processes in the network probe were switched off. The test was performed 3 times in 10 minute intervals. The results for all 3 intervals were very similar. For clarity, Figure 6 and Figure 7 show data from the third 10-minute interval of the test.

As expected, the maximum processing speed of incoming data using network interfaces was approx. 0.96 Gb/s (≈ 120 MB/s), as these are Gigabit interfaces. However, the CPU load throughout the testing

ranged from 20 to 25 %. As for the use of RAM, the load here was in the order of percent. It is therefore clear that the limiting hardware component is the network card itself, as the other components did not reach even half of their maximum load.

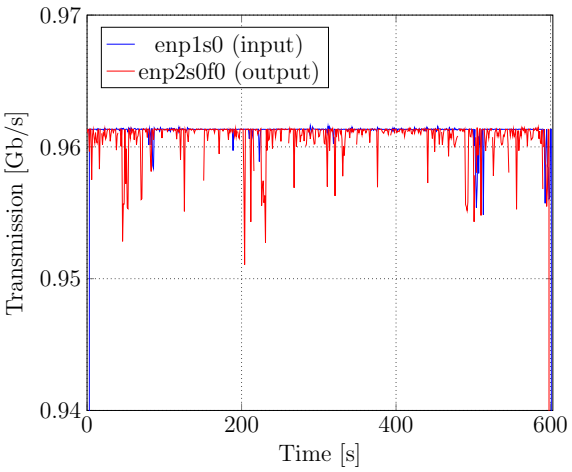

Figure 6: Usage of Ethernet interfaces during performance test

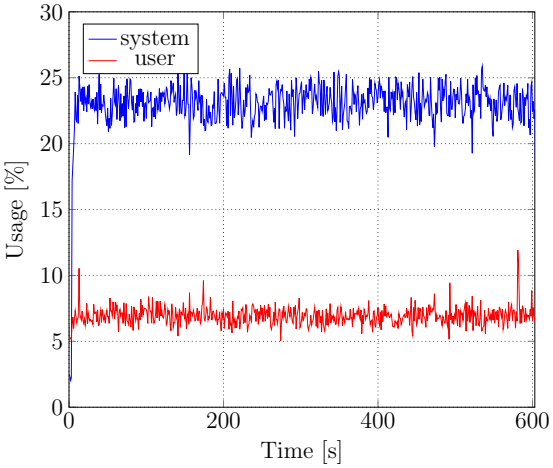

Figure 7: CPU usage during performance test

3.2. Rule detection test

The second set of tests focused on the ability of the network probe to evaluate the rules set in the Suricata module. As in the first set of tests, they focused on the usage of hardware components (CPU, RAM and network interfaces). Testing was performed on the schema shown in Figure 5. The network probe was set to the so-called brigde mode, which represents a transport connection, where traffic from one network interface is forwarded to another one.

The test set consisted of 3 times 6 measurements, where each measurement contained a different number of Suricata rules (1, 5, 100, 500, 1 000 and 5 000). Due to the larger number of tests, the time interval for this testing was reduced to 1 minute. Again, the most important parameters were the load on the network interfaces and the CPU load caused by the system. Due to the consistency of all recorded results and for clarity, only the data for the third set of tests for 1, 100 and 5 000 rules are shown in Figure 8 and 9. The results show that even for 5 000 rules, there is no noticeable increase in CPU usage. The average value ranged from 17 to 20 %, with a maximum of 22.5 %. As in the previous set of tests, the RAM load is in the order of percent. It is clear that the number of rules does not significantly affect the functionality of the probe.

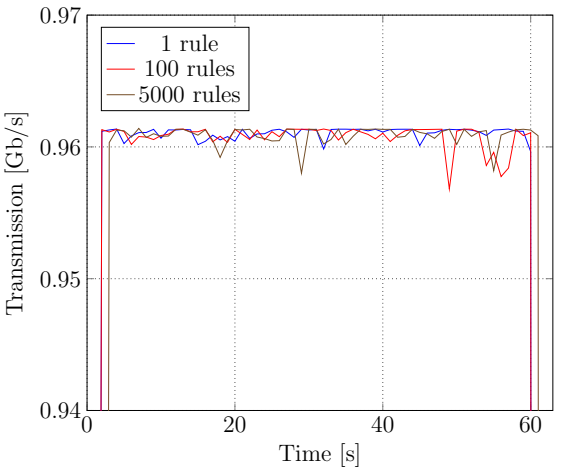

Figure 8: Usage of Ethernet interface *br0* during rule detection test

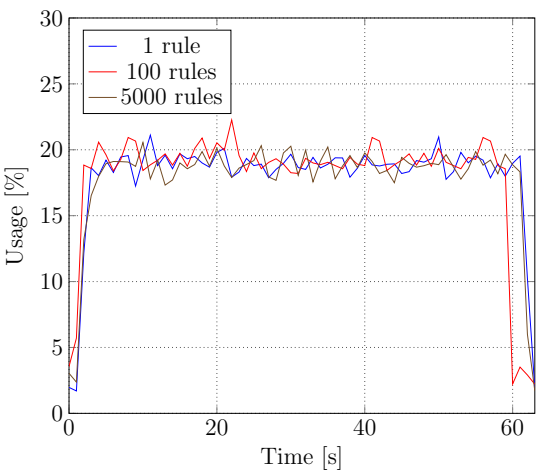

Figure 9: CPU usage during rule detection test

The delay and latency (so-called jitter) of the transmitted data passing through the network probe was also measured. The *Iperf* tool and the schema in Figure 5 were used for this testing. The results are displayed in Table II. The average delay caused by the network probe is around 0.007 ms. This value does not represent a significant deterioration in the connection properties. In the case of jitter, the values were 6 times higher when the mains probe was fitted. This is due to the different congestion densities of the network adapter when received packets are processed. A jitter value of 0.038 ms is not a problem for common networks.

Path	Minimum [ms]	Average [ms]	Maximum [ms]	Jitter [ms]
Not over network probe	0.167	0.184	0.251	0.007
Over network probe	0.217	0.257	0.410	0.038

Table II: Delay and jitter caused by the network probe

4. CONCLUSION

Nowasays, the issue of network security is a constantly discussed topic. On one hand, there is an effort to develop new technologies, but on the other hand, new technologies contain vulnerabilities that represent potentially unprotected areas of the system. To improve the security, for example in a corporate network, we can use the so-called network probes.This paper discusses the developed BUT network probe, which combines features of devices designed for analysis and filtering of network traffic. For easier user control, a web interface has been implemented in the network probe. This allows, among other things, setting up the network probe as IDS/IPS. For better analysis and recording of transmitted data, the T-shark module was implemented in the network probe, which creates a record of traffic in PCAP format. The last special feature of the network probe is the ability to analyze special protocols that are used in the industry, such as DLMS, IEC 60870-5-104, GOOSE and Sample Value protocols from the IEC 61850 standard. The paper also describes the tests used in verifying the functional characteristics and capabilities of the network probe. These tests have shown that the BUT network probe is capable of handling network traffic up to 1 Gb/s without significant degradation of traffic properties.Future work on the network probe will focus on expanding the supported industrial protocols, deploying external network cards for higher data rates using the USB-C interface, and adding new capabilities within the web interface.

REFERENCES

[1] H. Wang, B. Han, J. Su, and X. Wang, "A high-performance intrusion detection method based on combining supervised and unsupervised learning," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, 2018, pp. 1803–1810.

[2] I. Mukhopadhyay, M. Chakraborty, and S. Chakrabarti, "Hawkeye solutions: a network intrusion detection system," in *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, 2011, pp. 252–257.

[3] M. M. Aravind and V. Kalaiselvi, "Design of an intrusion detection system based on distance feature using ensemble classifier," in *2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*. IEEE, 2017, pp. 1–6.

[4] S. F. Diyar and A.-K. Samir, *Automatic Intrusion Prevention Technique to Improve Network Security: Intrusion Detection System (IDS), Intrusion Prevention System (IPS) and Firewall to secure the network*, 1st ed. LAP LAMBERT Academic Publishing, 2017.

[5] E. D. Knapp and J. Langill, *Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems*. Syngress, 2014.

[6] D. Hirš, "Proposal of cyber threat detector using raspberry pi," in *Proceedings I of the 27st Conference STUDENT EEICT 2021: General papers*. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2021, pp. 192–195. [Online]. Available: http://hdl.handle.net/11012/200741

[7] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Practical tools for attackers and defenders," in *Network Traffic Anomaly Detection and Prevention*. Springer, 2017, pp. 201–242.