A Review of Wazuh Tool Capabilities for Detecting Attacks Based on Log Analysis

Stefan Stanković, Slavko Gajin, and Ranko Petrović, *Member, IEEE*

***Abstract*— During the difficult times of the Covid pandemics and the transfer of work from the office to the home, security has never been more challenging. Because the development of information technology is expanding day by day, there is increasing amount of network traffic. Within that traffic, a potential attacker can often cover up his evil intentions. To detect attacks on host computer and prevent it from further malicious activities, Host Intrusion Detection Systems are often used. One of these systems is Wazuh and thanks to its powerful features it has been adopted by many companies. This paper provides an overview of the possibilities of Wazuh tools with a special emphasis on well-known attack detection on web servers.**

***Index Terms*—Wazuh, web-server, network, security, monitoring, attack.**

I. Introduction

The network monitoring system is used in the internal or external network in order to best identify risk components and prevent system crashes. The task of these systems is to find a weak point, submit a report and, if possible, solve the problem. Given the growing challenges in cyber security and the increasing amount of data generated globally, network and security administrators face a growing challenge. Most engineers use the Host Intrusion Detection System to detect and identify attacks and the Intrusion Prevention System to prevent them.

One of the main components of any application and an almost inevitable part of any data center is a web server - a hardware-software component that houses websites that serve end-users. Web servers are mostly exposed to the Internet and thus exposed to a large volume of potential attacks coming either from real attackers or from automated bots. Identifying attacks on web servers is a basic task of any administrator who maintains them because if protection is breached, the application may be inaccessible to a large number of users or permanently destroyed [1]. There are many network security monitoring solutions used worldwide [2]. One of the tools that helps identify and detect attacks on web servers is Wazuh. This paper will present the main features of the Wazuh tool installed in the University of Belgrade Computer Center.

Wazuh is a tool used around the world for various purposes. Paper [3] describes how the Wazuh tool can be used to test solutions that detect attacks within their constructed honeypot. On the other hand, Wazuh can integrate with machine learning as described in [4]. The advantage of this work is that it can serve network and security engineers very well in network and host security monitoring.

This work demonstrates Wazuh tools when collecting data exclusively from web servers. The results obtained through this paper give administrators an insight into what needs to be changed within their configurations in order to bring their servers and the entire infrastructure to the highest security level.

The paper is written in 4 major sections. After the introduction, Section 2 describes the basic functionalities of Wazuh tools and experimental setup, followed by Section 3, where statistic data are shown. Section 4 presents the results related to known attacks such as SSH (Secure Shell) brute force. The last section presents the main conclusions with ideas for future work.

II. Wazuh Overview and experiment setup

Wazuh is a free and open-source platform for threat detection and security monitoring according to predefined security rules. It can be used to monitor endpoints such as desktops, laptops, servers, or network devices such as firewalls and routers, and to aggregate and analyze data in real-time. Wazuh provides the following capabilities [5]:

- Security analytics - collection, aggregation, indexing and processing of security data, helping organizations detect intrusions, threats and behavioral anomalies.
- Intrusion Detection - Wazuh agents scan the monitored systems looking for malware, rootkits and suspicious anomalies. They can detect hidden files and processes.
- Log Data Analysis - Wazuh agents read operating system and application logs, and securely forward them to a manager for rule-based analysis.
- File Integrity Monitoring - Wazuh monitors the file system, identifying changes in content, permissions, ownership and attributes of files that need attention.
- Vulnerability Detector - Wazuh agents pull software inventory data and send this information to the server, where it is correlated with periodically updated CVEs (Common

Stefan Stanković is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia and with Vlatacom Institute oh High Technologies, 5 Milutina Milankovica, 11070 Belgrade, Serbia (email: stefan.stankovic@vlatacom.com).

Slavko Gajin is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (email: slavko.gajin@rcub.bg.ac.rs).

Ranko Petrović is with the Vlatacom Institute oh High Technologies, 5 Milutina Milankovica, 11070 Belgrade, Serbia (email: ranko.petrovic@vlatacom.com).

Vulnerabilities and Exposures) databases, in order to identify well-known vulnerable software.

- Configuration Assessment - Wazuh monitors system and application configuration settings to ensure they are compliant with security policies and standards. Agents perform periodic scans to detect applications that are known to be vulnerable, unpatched, or insecurely configured.

A. *Wazuh components*

The Wazuh solution is based on the following 3 components [6]:

- Wazuh agent - Installed on endpoints such as laptops, desktops, servers or virtual machines, it provides prevention, detection and response capabilities. It supports Windows, Linux, MacOS, HP-UX, Solaris and AIX platforms.
- Wazuh server - It analyses data received from the agents, processing it through decoders and rules, and using threat intelligence to look for well-known indicators of compromise.
- Elastic Stack -Elastic Stack is a unified suite of open-source projects for log management, including Elasticsearch, Kibana, Filebeat, and others. The projects that are especially relevant to the Wazuh solution are: Filebeat, Elastic Search, Kibana. Filebeat is A lightweight forwarder used to transfer logs across a network, usually to Elasticsearch. It is used on the Wazuh server to transfer events and alerts to Elasticsearch. Elastic search is A highly scalable, full-text search and analytics engine. A flexible and intuitive web interface for mining, analyzing, and visualizing data. It runs on top of the indexed content in an Elasticsearch cluster. Wazuh web user interface has been fully embedded in Kibana, in the form of a plugin. Wazuh architecture.

B. *Wazuh architecture*

The Wazuh architecture is based on agents, running on the monitored endpoints, that forward security data to a central manager. Moreover, agentless devices (such as firewalls, switches, routers, access points, etc.) are supported and can actively submit log data via Syslog. The manager decodes and analyzes the incoming information, and passes the results along to an Elasticsearch for indexing and storage.

C. *Experiment setup*

The results described in this paper were collected from several web servers with CentOS operating system version of 7.9, located in the University of Belgrade Computer Centre. Wazuh agents are installed on them to send Wazuh manager data. Wazuh manager has been installed on a virtual machine with Ubuntu operating system. Alternative to the implementation on virtual machine, dockers can be used, according to [7]. Some Wazuh manager functionalities are not included by default, such as Vulnerability Detector.

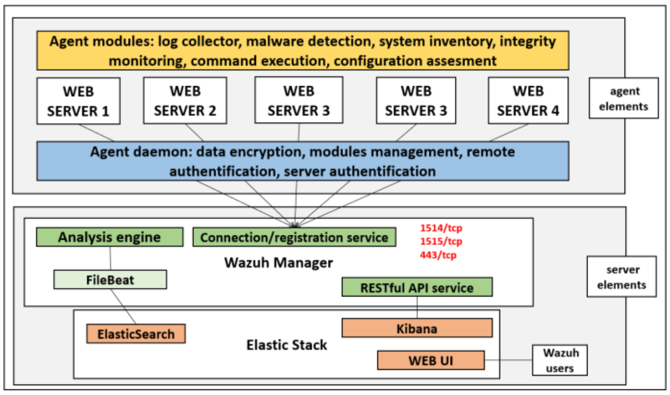

Agent modules: log collector, malware detection, system inventory, integrity monitoring, command execution, configuration assesment
WEB SERVER 1
WEB SERVER 2
WEB SERVER 3
WEB SERVER 3
WEB SERVER 4
agent elements
Agent daemon: data encryption, modules management, remote authentification, server authentification
Analysis engine
Connection/registration service
1514/tcp
1515/tcp
443/tcp
Wazuh Manager
FileBeat
RESTful API service
server elements
ElasticSearch
Kibana
Elastic Stack
WEB UI
Wazuh users

Fig. 1. Wazuh architecture on experiment setup

III. WEB SERVER ATTACK DETECTION OVERVIEW

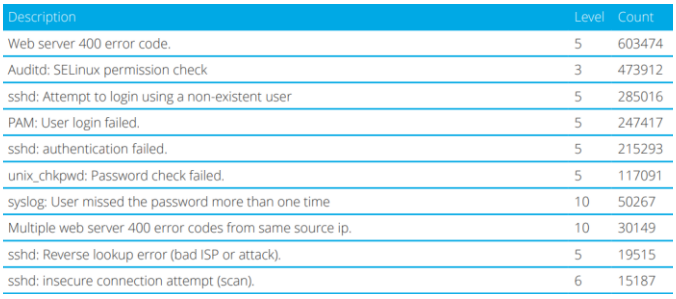
In this section the results within each element of Wazuh Managers will be presented in brief outlines and special emphasis will be placed on the analysis of well-known attacks. Within the main dashboard, there are 4 basic sections with options in which you can monitor data in real-time.

A. *Security Information Management*

Within this module there are 2 units in which statistics on security events and integrity monitoring are located.

1) *Security events*

In this section, it is possible to search for all security events recorded within the Wazuh system. The operation of the system is based on agents that send data (logs) to the server where they are processed. There is a whole set of rules defined to identify threats. The results are processed and when a rule is met then it is recorded within the dashboard. By default, the rules are divided into 12 levels based on defined standards. Wazuh provides the option to write custom rules according to user needs. Figure 2 shows the sorted list of security alerts. We see that the 'Web server 400 error code' is the most prevalent error. In each unit within the Wazuh manager, it is possible to display the results in a given time range. Within each section, there is an option to generate reports and for better visibility top 10 alerts from each report will be displayed. Figure 2 presents the top 10 alerts from 31-Mar-2022 to 27-Apr-2022.



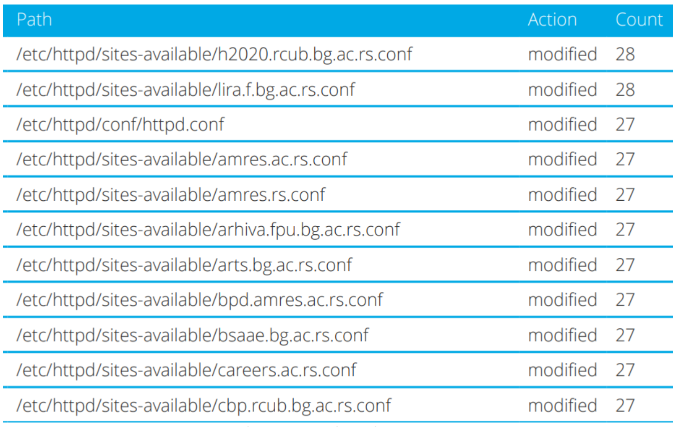
Description	Level	Count
Web server 400 error code.	5	603474
Auditd: SELinux permission check	3	473912
sshd: Attempt to login using a non-existent user	5	285016
PAM: User login failed.	5	247417
sshd: authentication failed.	5	215293
unix_chkpwd: Password check failed.	5	117091
syslog: User missed the password more than one time	10	50267
Multiple web server 400 error codes from same source ip.	10	30149
sshd: Reverse lookup error (bad ISP or attack).	5	19515
sshd: insecure connection attempt (scan).	6	15187

Fig. 2. Top 10 security alerts

2) *Integrity Monitoring*

In this module, it is possible to monitor the statistics of changes over system files on the host with the installed agent. These changes include modifying files, deleting files, and adding new files. Changes are detected based on the change in the checksum of each file.

Within the Events tab, it is possible to follow each change in detail, where you can find out the details of when a file was changed, which user did the action and what are the file permissions. This type of monitoring can be very useful, especially for sensitive systems. The performed actions and the list of files that are most often modified are shown in Figure 3.



Path	Action	Count
/etc/httpd/sites-available/h2020.rcub.bg.ac.rs.conf	modified	28
/etc/httpd/sites-available/lira.f.bg.ac.rs.conf	modified	28
/etc/httpd/conf/httpd.conf	modified	27
/etc/httpd/sites-available/amres.ac.rs.conf	modified	27
/etc/httpd/sites-available/amres.rs.conf	modified	27
/etc/httpd/sites-available/arhiva.fpu.bg.ac.rs.conf	modified	27
/etc/httpd/sites-available/arts.bg.ac.rs.conf	modified	27
/etc/httpd/sites-available/bpd.amres.ac.rs.conf	modified	27
/etc/httpd/sites-available/bsaae.bg.ac.rs.conf	modified	27
/etc/httpd/sites-available/careers.ac.rs.conf	modified	27
/etc/httpd/sites-available/cbp.rcub.bg.ac.rs.conf	modified	27

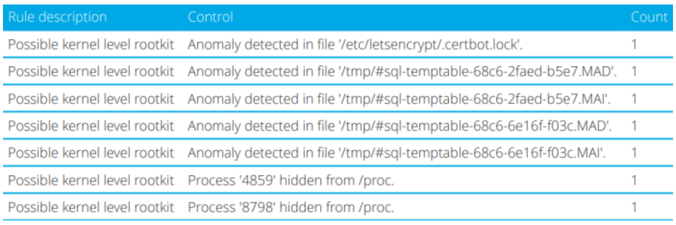
Fig. 3. The list of top modified files detected by Wazuh

B. *Auditing and Policy Monitoring*

This module offers 3 sections in which statistics and details about the system configuration and how much that configuration deviates from global standards.

1) Policy monitoring

This chapter shows the data obtained from the log analysis of policy monitoring. System configuration verification, such as kernel and security configuration files, is performed based on predefined rules. Wazuh uses 3 components to perform this task: Root check, OpenSCAP (Security Content Automation Protocol) and CIS-CAT. If some process is hidden from the virtual process file system (procfs), that file is marked as an alert.

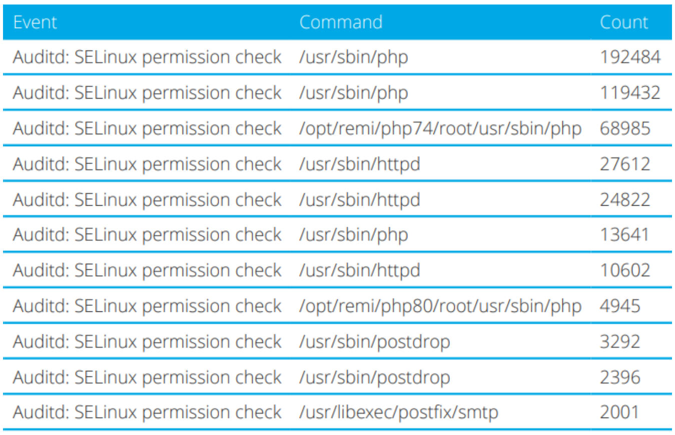


Rule description	Control	Count
Possible kernel level rootkit	Anomaly detected in file '/etc/letsencrypt/.certbot.lock'.	1
Possible kernel level rootkit	Anomaly detected in file '/tmp/#sql-temptable-68c6-2faed-b5e7.MAD'.	1
Possible kernel level rootkit	Anomaly detected in file '/tmp/#sql-temptable-68c6-2faed-b5e7.MAI'.	1
Possible kernel level rootkit	Anomaly detected in file '/tmp/#sql-temptable-68c6-6e16f-f03c.MAD'.	1
Possible kernel level rootkit	Anomaly detected in file '/tmp/#sql-temptable-68c6-6e16f-f03c.MAI'.	1
Possible kernel level rootkit	Process '4859' hidden from /proc.	1
Possible kernel level rootkit	Process '8798' hidden from /proc.	1

Fig. 4. The list of detected anomalies

2) System auditing

This section presents data based on which user behavior can be monitored, command execution monitored and possibly an alert raised if sensitive files are accessed. User behavior is monitored by a powerful auditing facility called *auditd* which provides a detailed accounting of actions and changes in a system. Thanks to the Wazuh agent who sends *auditd* logs to the manager, administrators can have insight into users' behavior. The statistics from this section are shown in Figure 5.



Event	Command	Count
Auditd: SELinux permission check	/usr/sbin/php	192484
Auditd: SELinux permission check	/usr/sbin/php	119432
Auditd: SELinux permission check	/opt/remi/php74/root/usr/sbin/php	68985
Auditd: SELinux permission check	/usr/sbin/httpd	27612
Auditd: SELinux permission check	/usr/sbin/httpd	24822
Auditd: SELinux permission check	/usr/sbin/php	13641
Auditd: SELinux permission check	/usr/sbin/httpd	10602
Auditd: SELinux permission check	/opt/remi/php80/root/usr/sbin/php	4945
Auditd: SELinux permission check	/usr/sbin/postdrop	3292
Auditd: SELinux permission check	/usr/sbin/postdrop	2396
Auditd: SELinux permission check	/usr/libexec/postfix/smtp	2001

Fig. 5. SElinux permission checklist

3) Security Configuration Assessment

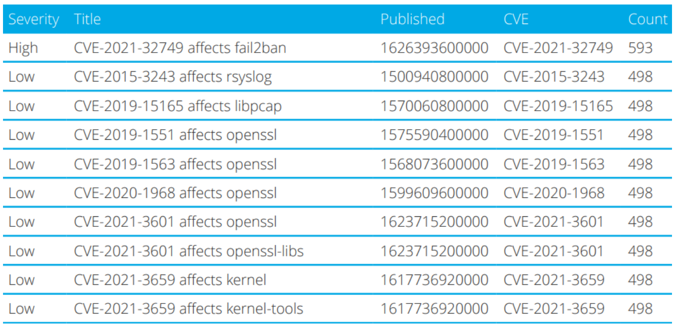
Within this unit, before displaying the data, it is necessary to select the agent where the configuration check will be performed. When the check is performed, the statistics and scores of that host are displayed. Verification is performed based on CIS (Center for Internet Security) benchmark recommendations for particular operating system distribution. The check consists of executing a set of commands whose result is binary: pass or fail. A detailed report is obtained when exported in CSV format where the columns show, among other things, commands, results, references and recommendations. This statistic gives a very good insight into the system configuration and draws the administrator's attention to important configuration elements.

C. *Threat Detection and Response*

In this unit there are 3 entities in which it is possible to gain insight into data related to threat detection. These entities are vulnerabilities, virus Total and MITRE ATT&CK, a globally accessible database of adversary tactics and techniques based on real-world observations. Evaluation of Wazuh tool with persistence tactic of MITRE ATT&CK is nicely described in [8]. As the targeted system is not related to antivirus software, no data has been collected.

1) Vulnerabilities

This module is not included in the main configuration file by default and needs to be enabled. It performs vulnerability searches according to the latest indexes that are updated in real-time with Canonical, RedHat and National Vulnerability databases. The detector on the manager inspects the list of installed applications periodically sent by the agents. Based on this list, search and verification processes are performed within the local database with the latest CVE elements (Common Vulnerabilities and Exposures). Alerts are generated when a CVE affects a package installed on one of the monitored servers. That package is then marked as vulnerable. There are 2 types of scanning: full and partial. A full scan is done the first time the Vulnerability Detector is activated and then each packet is scanned individually. Partial scanning is done when new packages are installed.



Severity	Title	Published	CVE	Count
High	CVE-2021-32749 affects fail2ban	1626393600000	CVE-2021-32749	593
Low	CVE-2015-3243 affects rsyslog	1500940800000	CVE-2015-3243	498
Low	CVE-2019-15165 affects libpcap	1570060800000	CVE-2019-15165	498
Low	CVE-2019-1551 affects openssl	1575590400000	CVE-2019-1551	498
Low	CVE-2019-1563 affects openssl	1568073600000	CVE-2019-1563	498
Low	CVE-2020-1968 affects openssl	1599609600000	CVE-2020-1968	498
Low	CVE-2021-3601 affects openssl	1623715200000	CVE-2021-3601	498
Low	CVE-2021-3601 affects openssl-libs	1623715200000	CVE-2021-3601	498
Low	CVE-2021-3659 affects kernel	1617736920000	CVE-2021-3659	498
Low	CVE-2021-3659 affects kernel-tools	1617736920000	CVE-2021-3659	498

Fig. 6. The list of matched CVEs detected by Wazuh

2) *MITTRE ATT&CK*

This feature allows the user to customize the alert information to include specific information related to MITRE ATT&CK techniques. MITRE ATT&CK matrix stores all possible attacks that can be made and what to do to detect them and mitigate the risk [9]. This can be useful when an attack is detected through an alert and a user wants to know more about it. MITRE ATT&CK assigns each attack technique an ID (identification). These techniques are grouped by tactics (Defense Evasion, Privilege Escalation, etc.) although some of them belong to more than one tactic.

D. Regulatory Compliance

The Wazuh platform is often used to meet the technical aspects of regulatory compliance standards. Wazuh not only provides the necessary security controls such as host intrusion detection, configuration assessment, log analysis, and vulnerability detection, among others, to meet compliance requirements but also uses its SIEM (Security Information and Event Management) capabilities to centralize, analyse and enrich security data. In order to provide regulatory compliance support, the Wazuh rules have been mapped against compliance requirements [10]. This way, when an alert is generated (a rule condition has been matched), it automatically includes compliance information. The following standards are supported: Payment Card Industry Data Security Standard (PCI DSS), General Data Protection Regulation (GDPR), NIST Special Publication 800-53 (NIST 800-53), Good Practice Guide 13 (GPG13), Trust Services Criteria (TSC SOC2), Health Insurance Portability and Accountability Act (HIPAA)

IV. EXPERIMENT, RESULTS AND SYSTEM PERFORMANCE

As already mentioned, one of the attacks analyzed in more detail is the SSH brute force attack. a brute-force attack is performed by an attacker submitting many passwords or passphrases with the hope of eventually guessing correctly. The attacker systematically checks passwords and passphrases from the database until the correct one is found. Also, if a user tries to connect via SSH using a random username, they will be considered to have attempted an attack and that alert will be recorded. Data processing on the Wazuh manager is in real-time and a potential attack is detected and alerted almost immediately. Table I provides very detailed information about the alerts when a potential attacker tried to connect as a non-existent user 'dunja'.

The basic geo locations based on the source IP address are followed by information about the user trying to connect and the name of the decoder that analyzes the data. This is followed by a description of the full log and an abbreviated name as well as the original log file location.

TABLE I
SSH failed login overview

Parameter	Data
GeoLocation.city_name	Belgrade
GeoLocation.country_name	Serbia
GeoLocation.location	{ "lon": 20.4721, "lat": 44.8166 }
GeoLocation.region_name	Belgrade
_id	s2K6-H8Bao9qD9NQsksr
_index	wazuh-alerts-4.x-2022.04.05
data.srcip	217.24.19.131
data.srcport	11847
data.srcuser	dunja
decoder.name	sshd
decoder.parent	sshd
full_log	Apr 5 07:57:36 wazuh sshd[233488]: Invalid user dunja from 217.24.19.131 port 11847
id	1649145457
input.type	log
location	/var/log/auth.log
manager.name	wazuh
predecoder.hostname	wazuh
predecoder.program_name	sshd
rule.description	sshd: Attempt to login using a non-existent user
rule.gdpr	IV_35.7.d, IV_32.2
rule.gpg13	7.1
rule.groups	syslog, sshd, invalid_login, authentication_failed
rule.hipaa	164.312.b
rule.id	5710
rule.level	5
rule.mail	FALSE
rule.mitre.id	T1110
rule.mitre.tactic	Credential Access
rule.mitre.technique	Brute Force
rule.nist_800_53	AU.14, AC.7, AU.6
rule.pci_dss	10.2.4, 10.2.5, 10.6.1
rule.tsc	CC6.1, CC6.8, CC7.2, CC7.3
timestamp	Apr 5, 2022 @ 09:57:37.807

A. System performance

The Wazuh manager described in this paper is installed on an Ubuntu virtual machine which is assigned 8GB of RAM (Random Access Memory), 4 cores and 30GB of storage. It aggregates data from a total of 5 servers in the network. Of the allocated resources, the system uses about 4.7GB of RAM, a negligible percentage of CPU load, and about 11G of storage is filled. The system uptime is 2 months.

V. Conclusion

The aim of this paper is to present the functionality of the Wazuh tool in detecting attacks demonstrated on web servers. Web servers are components that are very exposed to the Internet and if they are not well protected, they are very susceptible to attacks of various kinds. In order to prevent attacks, they must first be detected and that is why Host Intrusion Detection systems are used. One of the solutions that can help is Wazuh. It is a powerful tool that displays all detected attacks in great detail and in real-time. Although this paper demonstrates the analysis of detected attacks on web servers, Wazuh is a tool used to analyze attacks across the entire infrastructure. This paper presents the basic principle of operation of Wazuh tools based on the agent-manager system. Agents installed on hosts send the log data for processing to the manager. Statistics of different types of attacks are presented and special attention is paid to details concerning some well-known attacks such as SSH brute force. The attack was successfully detected and shown almost immediately. As further work, the installation of agents on all infrastructure devices is proposed, without limiting on the type of device. It would be desirable to integrate the Wazuh tool with an antivirus software in order to get deep inspection on viruses, worms trojans and other malicious content. Some security issues are most successfully detected by inspecting a server's actual network traffic, which is generally not accounted for in logs. This is where a Network Intrusion Detection System can provide additional insight into security. One of those systems is Suricata. Because Suricata is capable of generating JSON (JavaScript Object Notation) logs of events, it has very good integration option with Wazuh, so this is also a proposal to future work.

ACKNOWLEDGEMENT

This paper was done at Vlatacom Institute on project P158 and partially supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia (grant number 2022/200103).

References

[1] M. Moh, S. Pininti, S. Doddapaneni, T.S. Moh "Detecting Web Attacks Using Multi-Stage Log Analysis", *6th IEEE International Conference on Advanced Computing*,2016, doi: 10.1109/IACC.2016.141

[2] I. Ghafir, V. Prenosil, J. Svoboda, M. Hammoudeh, "A survey on Network Security Monitoring Systems", 4th International Conference on Future Internet of Things and Cloud Workshops, 2016, DOI: 10.1109/W-FiCloud.2016.30

[3] R. M. Muhammad, I. D. Irawati, M. Iqbal "Integrated Security System Implementation for Network Intrusion", *Journal of Hunan University*, vol. *48*, no. *6*, pp. *183-188*, June, 2021.

[4] O. Negotia, M. Carabas "Enhanced Security Using Elastic Search and Machine Learning", *Advances in Intelligent Systems and Computing*, vol. *1230*, July, 2020.

[5] Wazuh documentation overview, https://documentation.wazuh.com/current/, last visited 28.4.2022

[6] Wazuh documentation components, https://documentation.wazuh.com/current/getting-started/components/index.html last visited 28.4.2022.

[7] F. Mulyadi, L. A. Annam, R. Promya and C. Charnsripinyo, "Implementing Dockerized Elastic Stack for Security Information and Event Management", 5th International Conference on Information Technology, 2020. DOI: 10.1109/InCIT50588.2020.9310950

[8] J. Chandler, "Evaluating Open-Source HIDS with Persistence Tactic of MITRE att&ck", SANS Institute, 2021.

[9] Wazuh documentation, mitre, available at https://documentation.wazuh.com/current/user-manual/ruleset/mitre.html, last visited 29.04.2022.

[10] Wazuh documentation regulatory compliance, available at https://documentation.wazuh.com/current/getting-started/use-cases/regulatory-compliance.html, last visited 29.04.2022.