

Kostiantyn SAVCHUK<sup>1</sup>, Yurii LAKH<sup>2</sup>, Morika RUSINKO<sup>3</sup>

Opiekun naukowy: Elena NYEMKOVA<sup>4</sup>

DOI: <https://doi.org/10.53052/9788366249868.19>

## WYKRYWANIE LUK W PUNKTACH KOŃCOWYCH POPRZEZ SYMULOWANIE WSPÓŁCZESNYCH ATAKÓW

**Streszczenie:** Praca poświęcona jest analizie luk w punktów końcowych oraz metodom ich ochrony. Współczesne ataki były symulowane w wirtualnym laboratorium. Moduł oprogramowania do automatycznego wykrywania ataków został zaimplementowany przy użyciu wzorców zachowań napastników i wskaźników włamania. Moduł został pomyślnie przetestowany poprzez ponowne symulowanie ataków.

**Słowa kluczowe:** luki w punktach końcowych, wskaźniki włamania, wzorce zachowań atakujących, symulacja ataku, automatyczne wykrywanie ataków

## ENDPOINT VULNERABILITIES DETECTION BY SIMULATING MODERN ATTACKS

**Summary:** The study is devoted to the analysis of endpoint vulnerabilities and methods of their protection. The modern attacks were simulated in the virtual laboratory. The automated attack detection software module was implemented using attacker behavior patterns and indicators of compromise. The module has been successfully tested by re-simulating the attacks.

**Keywords:** endpoint vulnerabilities, indicators of compromise, attacker behavior patterns, attack simulation, automatic attack detection

### 1. Introduction

In today's information society, a huge amount of confidential information is stored digitally. The goal of attackers can be both violation of aspects of information

---

<sup>1</sup> Lviv Polytechnic National University, student of department of Information Technology Security, specialty: cybersecurity, kostiantyn.savchuk.mkbbi.2021@lpnu.ua

<sup>2</sup> PhD, Associate professor, Lviv Polytechnic National University, department of Information Security, yurii.v.lakh@lpnu.ua

<sup>3</sup> PhD, Associate professor, Lviv Polytechnic National University, department of Information Technology Security, morika.k.rusinko@lpnu.ua

<sup>4</sup> DSc, Professor, Lviv Polytechnic National University, department of Information Technology Security, olena.a.niemkova@lpnu.ua

protection, and disabling the equipment of work processes. Also, the purpose of some cybercriminals may be to damage the reputation of the organization. Such methods are often used in business to hurt competitors.

Endpoints are targeted frequently. An endpoint is any physical or virtual device that is connected to the network. In general, endpoints are: laptops, stationary personal computers, tablets, mobile gadgets, smart watches, printers, servers, ATMs, medical gadgets, virtual servers [1]. Recent studies have shown that 30% of violations are related to the installation of malware on endpoints.

The number of infections continues to grow, despite the fact that there are many endpoint protection products on the market. Traditional endpoint protection methods are outdated and fail to cope with increasingly sophisticated threats. An endpoint cyberattack is any form of malicious activity that targets endpoints [4]. The aim of the attacks is to gain illegal access to the system or to the system data. Modern cyberattacks are complex, multi-stage operations that require a lot of time and resources. The breakdown of computer systems can take place in different ways: from complex and rare attacks with the compromise of network components to technically primitive ones with the compromise of commercial correspondence [5].

## 2. Problem definition

Endpoint security is the practice of protecting endpoints from unauthorized entry. This practice aims to protect data and workflows during data processing, storage and transmission [2]. Specific points for the implementation of these practices are described in reference practices, such as, CIS Benchmark [3]. They provide advice on securing endpoints with different operating systems. Unfortunately, not all of these modern methods of end-to-end security settings are fully described in these documents. It should be noted that such solutions are usually expensive and have specific properties that are not always necessary.

To collect information both during attacks and during normal operational activities at the endpoint, so-called log files are used [6]. These files are automatically generated by the computer each time an event with a certain classification occurs on the network. The data aggregation is used to analyze log files. This is the process of collecting data and presenting it in a generalized format. This is an important step because the accuracy of data analysis largely depends on the quantity and quality of the data. It is important to collect high-quality, accurate data and a large enough amount to produce relevant results. Data aggregation is useful for investigating cybersecurity incidents, because it is possible to compile statistics on the actions of criminals and determine indicators of compromising the attack. Manual analysis of log files takes a lot of time. This leads to the analysis of the attack for a long time and the delay in obtaining the results of this analysis.

The purpose of the work is to develop a module for automatic search for indicators of compromise of attacks and patterns of attacker behavior based on the exploitation of vulnerabilities of the Linux operating system.

The following tasks are solved for this purpose:

- simulation of attacks using a virtual stand;
- determination of indicators of compromise and patterns of behavior of the attacker at the endpoint;

- development of an automatic software module for finding attacks based on the found indicators of compromise and patterns of behavior of the attacker.

### 3. The practical part

#### 3.1. Simulation of various types of attacks

The virtual stand was based on the virtual machine Metasploitable, which was used as the target machine when simulating the attacks, and Kali Linux, which was used as the attacker's machine, figure 1. Metasploitable logging was set up similarly to the infrastructure for monitoring by cybersecurity professionals. This was done in order to further analyze the log files, identify patterns of attacker behavior, find indicators of compromise and detect similar attacks in the future using the found indicators of compromise.

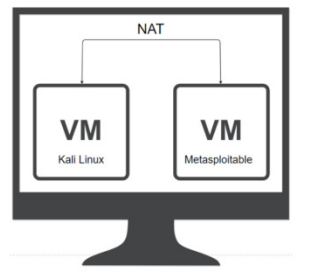


Figure 1. Scheme of a virtual stand

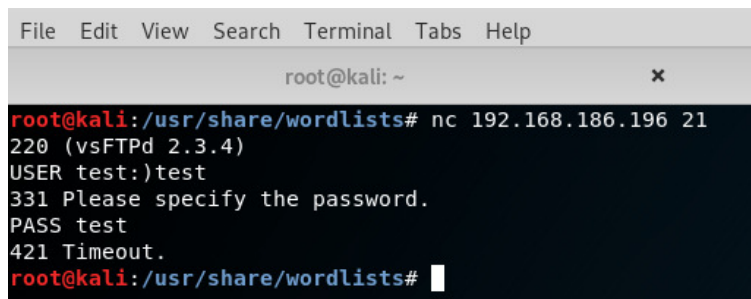
To simulate the attacks, the virtual lab was configured as follows. Firewall event logging has been extended to analyze the statistics of all network connections with the iptables `-A INPUT -j LOG` command. The login of the vsftpd service has been extended. vsftpd is the default FTP server on the Linux operating system. To extend logging, the vsftpd configuration file in the `/etc/vsftpd.conf` location was changed as follows: `“log_ftp_protocol = YES”` was added and `“xferlog_std_format = YES”` was commented out.

Four different types of attacks were simulated: scanning of open ports, exploitation of vulnerable software, brute force attack and attack of a man in the middle. Scanning of open ports of the target endpoint was performed using the utility nmap to start the attack and find potentially vulnerable points, figure.2.

```
root@kali:~# nmap -Pn -sC -sV -p$(cat ports.txt) -vv -oA nmap_ports_details 192.168.186.196
Starting Nmap 7.70 ( https://nmap.org ) at 2021-05-19 18:26 EDT
Scanning 192.168.186.196 [30 ports]
Discovered open port 111/tcp on 192.168.186.196
Discovered open port 80/tcp on 192.168.186.196
Discovered open port 445/tcp on 192.168.186.196
Discovered open port 53/tcp on 192.168.186.196
Discovered open port 22/tcp on 192.168.186.196
Discovered open port 21/tcp on 192.168.186.196
Discovered open port 25/tcp on 192.168.186.196
```

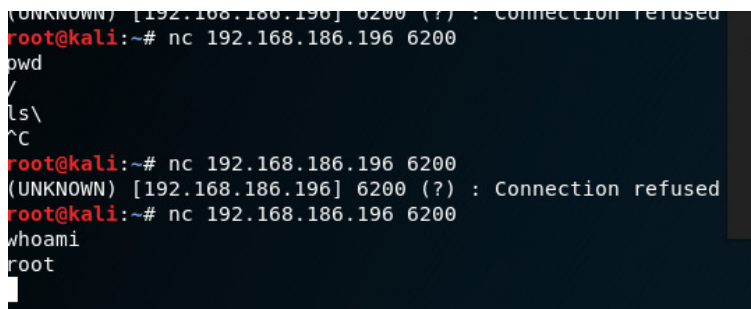
Figure 2. The result of a detailed scan of the ports

The search for vulnerabilities in these services began after the end of the exploration phase of open ports and services. Since the vsftpd service version 2.3.4 was found on port 21, the decision to use the Metasploit platform was made to find the appropriate exploit. An exploit named “exploit/unix/ftp/vsftp\_234\_backdoor” was found. This version of vsftpd is vulnerable because someone placed the code in the vsftpd repository, which opens a backdoor when a smiley is used in the username :). This causes a backdoor in port 6200. After configuring the target machine, the exploit was executed and remote access to the endpoint was obtained with administrator privileges. The connection to port 21 was made using the username “test:)test” and the password “test”. Immediately after that, in another tab of the command line, the connection to port 6200 was performed and the “whoami” command was run to verify the user name, figure 3a, 3b.



```
File Edit View Search Terminal Tabs Help
root@kali: ~
root@kali:/usr/share/wordlists# nc 192.168.186.196 21
220 (vsFTPd 2.3.4)
USER test:)test
331 Please specify the password.
PASS test
421 Timeout.
root@kali:/usr/share/wordlists#
```

Figure 3a. Exploitation of the vsftpd vulnerability in manual mode



```
(UNKNOWN) [192.168.186.196] 6200 (?): Connection refused
root@kali:~# nc 192.168.186.196 6200
pwd
/
ls\
^C
root@kali:~# nc 192.168.186.196 6200
(UNKNOWN) [192.168.186.196] 6200 (?): Connection refused
root@kali:~# nc 192.168.186.196 6200
whoami
root
```

Figure 3b. User verification for remote access

A weak password was the next vulnerability that was exploited. During the scan of open ports, port 22 was detected, which is responsible for ssh (secure shell) connection to the endpoint. The brute force technique was used to access the endpoint using the Hydra utility. Hydra is a brute force attack utility that supports a large number of protocols and conducts parallel testing. Also, pre-configured dictionaries were used to form a login-password pair for brute force. After starting the utility found the login and password for remote access to the endpoint via ssh, figure 4.

```
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-05-19 19:03:48
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ssh://192.168.186.196:22/
[22][ssh] host: 192.168.186.196 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-05-19 19:03:49
root@kali:~/usr/share/wordlists#
```

Figure 4. Successful brute force login and password

The entry to the end point was made after a successful attack of searching for logins and passwords. An attempt to obtain administrator rights was successful.

The attack of the man in the middle was the next attack, which was simulated. During this attack, the connection to the endpoint via telnet was made. The telnet protocol is vulnerable to attacks of this type because it is not secure and does not use data encryption. The Wireshark program was used to intercept and analyze traffic. During the analysis of the intercepted packets, the login and password were found in the open, figure 5.

```
metasploitable login: mmsfffaaddmminn
Password: msfadmin
```

Figure 5. Login and password in intercepted packets

Simulation of all attacks was successful and as a result the endpoint was accessed by three different methods.

All attacks were simulated according to the most popular frameworks Mitre Matrix and Kill Chain. Comparisons of these frameworks in relation to the attacks are given in the table 1.

Table 1. Correspondence of simulated attacks to well-known tactics, techniques and phases of attacks

Attack	Tactics Mitre	Technique Mitre	Phase Kill Chain
Scanning open ports	Reconnaissance, Discovery	Active Scanning, Network Service Scanning	Reconnaissance
Operation of vulnerable software	Initial Access	Exploit Public-Facing Application	Exploitation
Bruteforce attack	Credential Access	Brute Force	Reconnaissance
Attack Man-in-the-Middle	Credential Access, Collection	Man-in-the-Middle	Reconnaissance

### 3.2. Indicators of compromise

Attack analysis usually begins with system log files that contain records of endpoint network connections. This is the best way to start an investigation, because each attack begins with the reconnaissance phase - finding open ports at the end point. The transition to the directory with log files was performed. The log file “/var/log/syslog” was filtered using the following command: “cat syslog | grep IN=”. The first part of the command displays the contents of the file on the screen, and the second filters the lines of the file. In this case, the filtering was performed according to the pattern “IN =”. It is from this pattern that the Linux firewall logs begin.

After that, a review of the logs was carried out and a certain pattern was found. It consisted of a large number of records that contained the same values of certain fields. For example, the query length displayed in the “LEN” field was the same in all records. Also, all requests displayed in the logs were made using the TCP protocol, which provides for the so-called handshake, but the handshake itself was not completed, which is a sign of scanning ports with the nmap utility with the -sS parameter. Moreover, all requests came from a single port 36617 and were made from the same IP address 192.168.186.195.

The next step was to modify and further filter the contents of the log file using the commands “cat syslog | grep 192.168.186.195 | grep IN = | head -3” and “cat syslog | grep 192.168.186.195 | grep IN = | tail -3”. These commands filter the logs according to the specified IP address and output the first and last three records with this address, respectively. Thus, it was possible to identify and confirm the following findings and evidence of the attack, table 2.

*Table 2. Findings and evidence of the attack*

Scan start time	May 19 15:34:29	Defined by entries in the log file
Scan end time	May 19 15:34:39	Defined by entries in the log file
Scan duration	10 seconds	Calculated manually
IP address of the attacker	192.168.186.196	Defined by entries in the log file
Scan type	The nmap utility with the -sS parameter	Assumptions made on the basis of findings

Further analysis of utility and application files was performed in order to identify further actions of the attacker.

The IP address of the attacker was detected in the file “vsftp.log” using the command “sudo cat <filename> | grep 192.168.186.195”, where <filename> was replaced alternately by the name of each log file. This indicates that an attacker will attempt to connect to an FTP server.

Further review of the records revealed attempts to enter the attacker with logins that contained “:).” Knowing the possible vulnerability of the installed version of the FTP server, we can say with confidence that it was exploited by an attacker. It should be noted here that untimely software updates often lead to such attacks. Software manufacturers recommend that you install updates as soon as they are published. This makes it possible to avoid similar situations.

A similar method of searching for the attacker's IP address in the log files, another log file called “auth.log” was detected. This file stores data about login attempts to the endpoint. After a detailed review of the file, a large number of unsuccessful attempts to enter the endpoint using the SSH protocol were noticed. Also in the log file you can see that after unsuccessful attempts, the attack was successful. These facts indicate the success of the attack called a brute force attack, or brute force, table 3.

*Table 3. Brute force attack*

Scan start time	May 19 17:57:38	Defined by entries in the log file
Scan end time	May 19 18:34:26	Defined by entries in the log file
Scan duration	36 minutes and 48 seconds	Calculated manually
Time of successful login	May 19 18:34:26	Defined by entries in the log file
Compromised account	msfadmin	Assumptions made on the basis of findings

### 3.3. Automation of attack detection

Python programming language was used to implement the automatic attack detection module, this language works by default in the Linux operating system and does not require any additional configurations.

To write a program, the following sequence of actions was taken, which the program must perform:

- Get login, password and IP address of the endpoint.
- Download log files from the endpoint for further local analysis.
- Parsing and formatting log files for easy analysis.
- Find indicators of compromise and patterns of behavior of the attacker.
- Give general statistics on the attack, or state the absence of indicators of compromise, and hence the attack itself.

Therefore, to detect an attack on a vulnerable component vsftpd, it was decided to use the login as an indicator of compromise. After all, thanks to the special characters in the login, the attack will be successful. And the vulnerability is precisely the use of a sequence of characters “:”)” for the attack. Accordingly, all usernames that were used when logging in to the endpoint and that contained these characters can be considered as used by an attacker.

Next, a module for detecting open port scans was developed. The following behavior pattern is used: a large number of different ports that have been attempted to connect to another endpoint in a short period of time. The use of the same source port was also added to this pattern. This pattern was noticed during a manual attack analysis.

The next module is a module for detecting a brute force attack. The pattern is very similar to what was used in writing the previous module. Namely: a large number of unsuccessful login attempts in a short period of time. As a result, the program successfully detected all three attacks, as well as provided useful statistics on these attacks.

To test the program, it was necessary to simulate the attacks again. To do this, the Metasploitable virtual endpoint was returned to the so-called “snapshot”. That is, in a state exactly as it was before the simulation of attacks. The program was then run to look at the results of its execution in the absence of indicators of compromise in the log files. As a result, the program worked without errors, and gave correct results, which confirmed the absence of attacks on the endpoint, figure 6.

```

root@kali: ~
root@kali:~# python attack_finder.py
This program check log files of target machine for any indicators of compromise.
Thanks to that following attacks can be easily detected:
vsftpd Exploitation
Brute Force
Open Ports Scanning
Please enter login of target machine: msfadmin
Please enter password of target machine: msfadmin
Please enter IP address of target machine: 192.168.186.196
There was no attacks on vsftpd
There was no port scans
There was no brute force attacks
root@kali:~#

```

Fig.6. Operation of the program after restoring the attacked machine to “snapshot”

After simulating the attacks, the program was run again to get information about the attacks. Attack statistics have changed, namely: attack time and source port have changed. This indicates that the program is working properly and that it is able to find three types of attacks. Namely: exploitation of the vsftpd vulnerability, brute force attack and scanning of open ports.

#### 4. Acknowledgments

The research had been performed in the framework of International Project of CRDF Global “Developing software and hardware complex for dynamical authentication of information processing devices in a corporate network for cybersecurity purposes”, supported by the U.S. Department of State, the Bureau of European and Eurasian Affairs. Grant Agreement: G-202102-67366.



#### 5. Conclusion

Manual analysis of attacks on endpoints was performed. As a result, patterns and indicators of compromise were detected, which were then used in the program to automatically detect these attacks. Three attacks were detected, namely: open port scanning, brute force, and exploitation of vulnerable software. The attack of the man in the middle was not detected due to the lack of his traces at the end point. This attack is best detected using network equipment log files, which contain records of packets being sent between all points on the network, as well as records of devices connected to that network.

The program is implemented in Python and tested in the following ways: testing on log files from previous attacks, testing on log files without traces of attacks and testing



on log files generated during re-simulation of attacks. The program was proven to work as a result of multiple simulations and the extent to which it speeds up the analysis of attacks.

Further research will focus on expanding the range of endpoint attacks.

## LITERATURA

1. Endpoint Security: <https://www.cisco.com/c/en/us/products/security/endpoint-security/index.html>, 06.05.2021
2. What is a Cybersecurity Attack?: <https://intsights.com/glossary/what-is-a-cybersecurity-attack>, 06.05.2021
3. What are the Most Common Cyber Attacks?: <https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html>, 07.05.2021
4. What Is Endpoint Security?: <https://www.mcafee.com/enterprise/en-us/security-awareness/endpoint.html>, 08.05.2021
5. CIS Benchmarks: <https://www.cisecurity.org/cis-benchmarks/>, 08.05.2021
6. Security log: [https://en.wikipedia.org/wiki/Security\\_log](https://en.wikipedia.org/wiki/Security_log), 08.05.2021

