Journal of Computing Research and Innovation (JCRINN) Vol. 7 No. 1 (2022) (pp155-163)
https://jcrinn.com : eISSN: 2600-8793 doi: 10.24191/jcrinn.v7i1.282

Integrated Network Monitoring using Zabbix with Push Notification via Telegram

Mohd Faris Mohd Fuzi[1]*, Nur Fatin Mohammad Ashraf[2], Muhammad Nabil Fikri Jamaluddin[3]
[1,2,3] Faculty of Computer and Mathematical Sciences,
Universiti Teknologi MARA Perlis Branch, Arau Campus, 02600 Arau, Perlis, Malaysia

*Corresponding author: * farisfuzi@uitm.edu.my*

Received Date: 16 January 2022
Accepted Date: 29 March 2022
Published Date: 30 March 2022

HIGHLIGHTS

- Continuous ICMP logged by tcpdump is identified as a possible ping flooding attempt.
- Continuous SYN packets from the same IP with no SYN_RCV reply is used as the parameter to trigger an alert to the administrator.
- Alerts of problems and resolved problems are sent to the administrator for both ping and SYN flooding attempts.
- Zabbix requires at least a 30 second interval for data retrieval from the monitored host.

ABSTRACT

The world is becoming increasingly dependent on online services. To offer a service, a network must be in good health and free of any attacks. An attack happens when the confidentiality, integrity, or availability of a service is compromised. Network monitoring is a solution capable of maintaining these network devices from their usage up to detecting attacks. A denial of service (DoS) attack on a network can affect the network performance and can cause serious damage. Zabbix is an open-source network monitoring tool that is versatile and can be used to monitor hosts on a network. The purpose of this project is to detect possible ping and SYN flooding attempts on a server and send alerts to the administrator via Telegram. This project uses Zabbix to monitor a server for potential ping and SYN flooding attacks. Tcpdump is used to log the pings received by the server. When the server continuously receives 10 or more pings per second, an alert will be automatically generated and sent to the administrator via Telegram. Similarly, a SYN flood attack is detected by using netstat's SYN_RECV flags. When the server continuously receives more than 10 SYN packets without an ACK packet, Zabbix will generate alerts that are sent via Telegram and update the dashboard to show a problem. Zabbix was able to accurately detect all ping flooding attempts on the server. However, SYN flooding attacks were not as accurately detected. The use of Zabbix can be implemented in small businesses or networks for an automated monitoring system. Future work can include more DDoS attacks and adding countermeasure actions when detecting attacks by blocking the IP or port associated with the attack. SYN flooding detection needs to be improved because only two out of three attacks were able to be caught.

Keywords: *Network Monitoring, Zabbix, Ping Flood, SYN Flood, Telegram*

INTRODUCTION

The Internet works by connecting millions of users around the world, enabling them to communicate and essentially exchange data with each other, making it the perfect example of a network. A threat is defined by the prospect of leveraging a security breach or vulnerability and thereby causing potential harm (Birkinshaw, Rouka & Vassilakis, 2019). Network security is a combination of multiple layers of defences at the edge and within the network. Automation in network security is the process of using software to automate network and security provisioning and management. Any attack that compromises confidentiality, integrity, and availability is considered an intrusion. There are several ways to identify if your network is vulnerable to attacks. This includes using existing tools like Tcpdump and Wireshark to monitor your network. However, running these tools separately is time-consuming, and manually analysing the results can be prone to human error as the administrator might miss an event. According to Nobles (2018), human-enabled errors account for 95% of the increasing cyber-attacks, data breaches, and ransomware attacks. A denial of service (DoS) attack on a network can affect the network performance and can cause serious damage. A DoS attack is difficult to avoid even as multiple techniques are adopted, as the attack can be implemented in several different ways (Abid, 2020). A ping flooding attack is a volume-based attack on a network by sending a large number of ICMP messages to the victim to consume the server's resources and deny access to legitimate users. SYN flooding is an attack that tries to make a server unusable for normal traffic by using all the server's resources at the same time.

RELATED WORKS

Hakim, Rinaldi, and Setiadji (2020) successfully implemented Snort as the NIDS and used WhatsApp and Telegram as the mediums through which the administrator receives an alert in their study. Snort was used as a signature-based NIDS in the study. On the victim machine, three types of attacks were carried out: a ping of death attack, a SYN flood attack, and an SSH Brute Force attack. The results show that the system implemented can detect simulated attacks and send notifications to the network administrator.

Johnson and Elizabeth (2018) conducted research that uses Nagios to notify the network administrator of a failure in the network topology via e-mail. Barbu, Pascariu, Bacivarow, Axinte, and Firoiu (2017) used Python to detect intruders on a local network. They can run a scan across a network and send alarms to the owner of any intrusions found using open-source software and low-cost hardware. Mardiyono, Sholihah, and Hakim (2020) used Zabbix and Telegram to successfully implement a mobile-based network monitoring system. During the research, the system was able to send warnings detected by Zabbix to the administrator via Telegram.

Sulistya and Sasmita (2020) created a monitoring system with notification alerts using Snort and Telegram. To test the notification system, they launched a UDP DoS attack and a Nmap port scan. Gayathri and Neelanarayanan (2018) conducted a study on a DoS detection solution for cloud platforms using SNMP that was successful in detecting and distinguishing between a DoS attack and legitimate requests by looking at the number of requests sent per second.

METHODOLOGY

The methodology used in this project consists of six phases, starting with the initial phase. The gathering and reviewing of information relevant to the project were reviewed. Reading materials and research on topics related to network monitoring, automation, and DoS attacks are used to gain a better understanding of the project. Using the information obtained, the background of the study was written. Information from the background study was then used to identify the project objective, problem statement, research scope, and research significance. The planning phase consists of using the information gathered during the initial phase to plan out the flow of the project. In this phase, the hardware and software requirements are identified.

The third phase is the design and development phase. During this phase, the network topology of this project is designed. Then, the scripts for collecting data and detecting ping and SYN flooding attacks are written. The dashboard configurations on Zabbix and setting up a Telegram bot were also done. In this phase, the network topology is also determined. Figure 1 illustrates the network topology of this project. To achieve the objectives, a ping flood attack and a SYN flood attack will be conducted on the victim server. A ping flood attack on the victim server will be launched using the ping command to send 10 pings per second. A bash script was implemented to execute tcpdump with cron logs the pings received on the victim server in tcpdump.log. If the server receives ten or more pings per second from the same source, it will be considered a possible ping flood. Next, a second script will be implemented to monitor the SYN_RECV socket status via netstat. To launch a SYN flood attack, hping3 will be used.

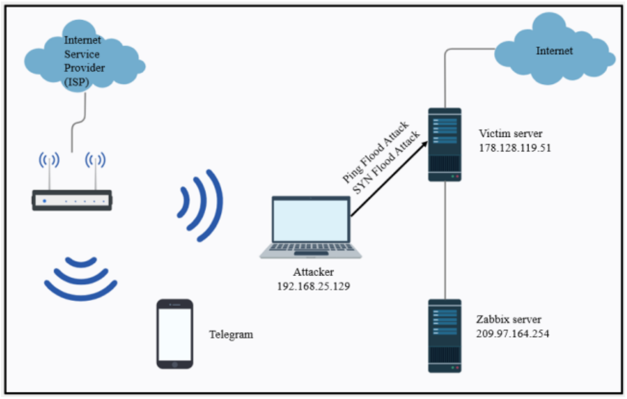


Figure 1: Network Topology

The next phase is the testing phase. In this phase, the system's capabilities alert the administrator of possible ping and SYN flooding attacks based on the triggers that were predefined during the design and development phase. There will be three tests and objectives to achieve in this phase. The objective of Test 1 is to test the ability to detect a ping flooding attack. In the first test, the parameters used to identify a possible ping flood attack are 10 consecutive pings from the same IP address per second. A study on DoS detection solutions for cloud platforms using SNMP by Gayathri and Neelanarayanan (2018) uses the assumption of 3 requests originating from the same source per second and is assumed to be a DoS attack. In this test, a machine will send 10 pings per second using the ping command. The expected outcome of this test is to analyze the graphs produced by Zabbix and the pings received in the tcpdump logs. A ping flood would cause a sudden spike in the graph. The second test will be conducted by conducting a SYN flooding attack. In this test, the trigger for a SYN flood attack is set to IPs with 10 or more SYN packets sent with no ACK packet. A SYN flood attack can be simulated using hping3. Next, the system's alert capabilities during a ping flood attack and a SYN flood attack are tested. To test the alert capability, a ping flood attack and a SYN flood attack will be conducted.

The fifth phase is the analysis phase. The time the attack started and stopped is recorded, and the time the administrator receives an alert is also recorded. The analysis done is on the accuracy of detection of the ping and SYN flooding attacks and the capability to alert the administrator of the attack. The final phase is documentation. All the steps taken to complete this project are recorded and documented.

Setting up

On DigitalOcean, two new droplets were created to host the servers in the cloud. The *V.Server* is the victim server with an IP address of 178.128.119.51, and the *ZABBIX-Server* is the Zabbix server with an IP address of 209.97.164.25. To install Zabbix, access to the server via putty is required. Zabbix's latest repository was installed from the Zabbix Official Repository. Then, the Zabbix server and web frontend with a MySQL database are installed using the *sudo apt install zabbix-server-mysql* and *zabbix-frontend-php* commands. The next step is to install the Zabbix agent on the victim server, which will collect data and send the data to the Zabbix server with *sudo apt install zabbix-agent*. To configure the Zabbix frontend, navigate to *http://209.97.164.254/zabbix* and review the prerequisites. Login to Zabbix with the default user credentials and make the necessary password change. Next, active checks enable Zabbix to collect data for it to monitor the victim server. The trappers are set to enable the processing of active checks on Zabbix with the command by starting the trappers in */etc/zabbix/zabbix_server.conf*. To receive notifications from Zabbix, a Telegram bot is needed. The bot will send alerts to the administrator. In the Zabbix frontend, a media type has been added. In Users, update the send to field and place the bot ID to start receiving notifications.

Launching Ping Flooding Attack

One of the attacks launched on the victim server to test the objective of the project was achieved. To launch a ping flooding attack, open the terminal and use the ping command with the *-n* and *-i* options. The *-n* option will show the IP instead of the hostname, and the *-i* option specifies the interval between successful packet transmissions. To simulate a ping flood attack that would be able to be detected, 10 or more ping requests need to be sent per second. The *-i* option will allow this by setting it to 0.1. Using the command *ping -n -i 0.1* , 10 pings will be sent per second to the target. This command, however, needs to be done as a root user by implementing the *sudo su* command. This would fulfil the trigger requirements for Zabbix to show the pings as a possible ping flooding attack.

Launching SYN Flooding Attack

A SYN flooding attack can be simulated using the hping3 tool available on Linux distributions. To install hping3 on Ubuntu, run the commands *sudo apt update* and *sudo apt install hping3*. To start the attack, use the command *hping3 --flood --rand-source --destport --syn -d 120 -w 64*. Based on the triggers set, any IP that sends 10 or more packets with the SYN flag on and receives the SYN+ACK packet from the target without sending an ACK back is considered an attempt at SYN flooding. To check if an attempted connection was a potential SYN flood attack, netstat was used to check the socket status for the SYN_RECV status. This indicates that the server has received the initial SYN packet, has sent its own SYN+ACK packet, and is waiting for the ACK from the attacker machine to complete the three-way handshake. When a server receives a large number of connections, it waits for a response from the client, leaving the connections open and draining the server's resources, resulting in a denial of service (Salunkhe, Jadhav & Bhosale, 2017).

Figure 2 shows the testing framework used in this project. Three tests with different objectives were conducted.

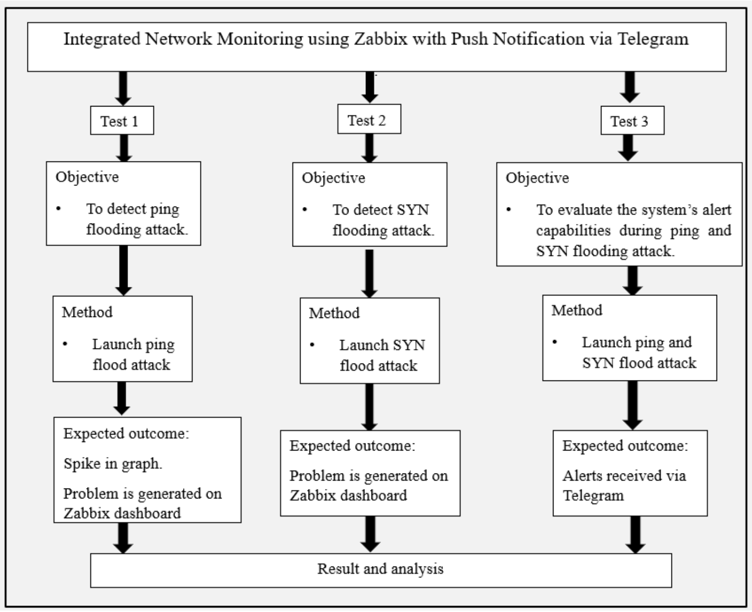

Integrated Network Monitoring using Zabbix with Push Notification via Telegram
Test 1
Test 2
Test 3
Objective
• To detect ping flooding attack.
Objective
• To detect SYN flooding attack.
Objective
• To evaluate the system's alert capabilities during ping and SYN flooding attack.
Method
• Launch ping flood attack
Method
• Launch SYN flood attack
Method
• Launch ping and SYN flood attack
Expected outcome:
Spike in graph.
Problem is generated on Zabbix dashboard
Expected outcome:
Problem is generated on Zabbix dashboard
Expected outcome:
Alerts received via Telegram
Result and analysis

Figure 2: Testing Framework

The first test involves conducting a ping flooding attack to observe a spike in the graph and the problem generated on the Zabbix dashboard. The second test was conducted by conducting a SYN flooding attack, and the expected outcome is that the problem will appear on Zabbix. The third test is to evaluate the system's alert capabilities during ping and SYN flooding. This test is done by launching the attacks and recording the time of the attack start and the time the administrator receives an alert.

FINDINGS AND DISCUSSIONS

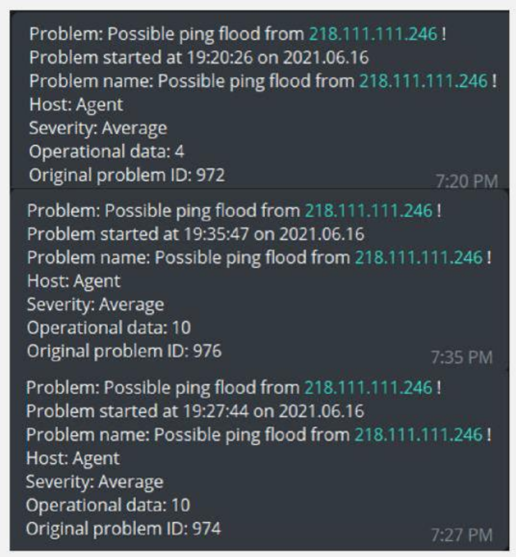
Ping Flooding Attack Results

Based on Table 1, three ping flooding attempts were launched to the victim server (178.128.119.51). Zabbix was able to detect these attacks using the trigger of 10 or more pings from the same source per second. The administrator receives an alert of the attack immediately after it appears on Zabbix's dashboard.

Table 1: Ping Flood Testing Results

No	Start Time of Attack (24hrs)	Alert Trigger	Problem listed on Zabbix Dashboard	Administrator's Alert (24hrs)	Time Attack is Stopped (24hrs)	Time Problem is Recovered (24hrs)
1	19:20	PROBLEM	Possible pin flood from 218.111.111.246 !	19:20	19:23	19:24
2	19:27	PROBLEM	Possible pin flood from 218.111.111.246 !	19:27	19:30	19:31
3	19:35	PROBLEM	Possible pin flood from 218.111.111.246 !	19:35	19:40	19:41

The alert sent to the administrator includes the following: the problem, the time the problem started, the problem name, which host the problem is on, the severity, the operational data, and the original problem ID. Once the attack has been stopped, Zabbix also sends the administrator an alert that the problem has been resolved. Based on the results obtained, Zabbix is successful in detecting the attacks in a timely manner and all alerts were received by the administrator as seen in Figure 3.



Problem: Possible ping flood from 218.111.111.246 !
Problem started at 19:20:26 on 2021.06.16
Problem name: Possible ping flood from 218.111.111.246 !
Host: Agent
Severity: Average
Operational data: 4
Original problem ID: 972
7:20 PM

Problem: Possible ping flood from 218.111.111.246 !
Problem started at 19:35:47 on 2021.06.16
Problem name: Possible ping flood from 218.111.111.246 !
Host: Agent
Severity: Average
Operational data: 10
Original problem ID: 976
7:35 PM

Problem: Possible ping flood from 218.111.111.246 !
Problem started at 19:27:44 on 2021.06.16
Problem name: Possible ping flood from 218.111.111.246 !
Host: Agent
Severity: Average
Operational data: 10
Original problem ID: 974
7:27 PM

Figure 3: Alerts Received by the Administrator in Telegram During Ping Flood Attack

During an attack, the graph shows the pings received are clustered together and is highlighted with a red hue. Zabbix uses the ICMP ping logged in tcpdump.log to map out the graph where each dot represents a ping received. The Zabbix dashboard illustrates the ping flood attacks launched on the ICMP ping requests from different sources (per 1 sec) graph where the pings received were clustered together. Based on Figure 4, when the victim server was attacked, the graph showed an increase or spike in the dots representing the pings it was receiving.

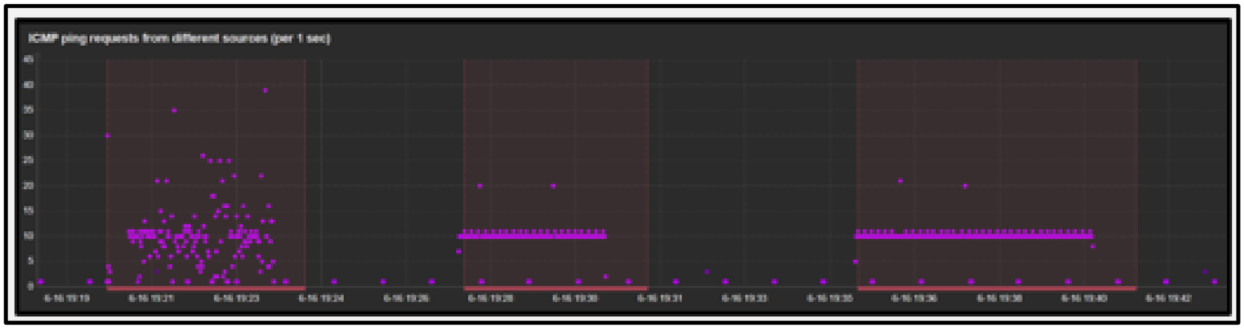


Figure 4: ICMP Ping Graph

SYN Flooding Attack Results

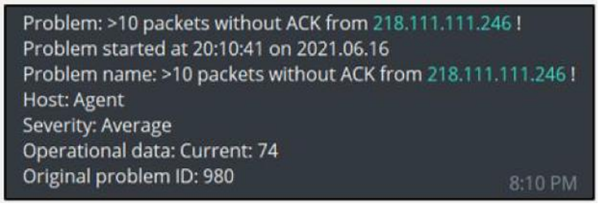
Table 2 shows the results obtained during the SYN flooding attack testing. Three SYN flooding attacks were launched using the hping3 *178.128.119.51 --c 5 --destport 22 --syn -d 120 -w 64* command. Zabbix was successful in detecting two of the attacks and alerts were sent to the administrator. Without stopping the attack, the victim server was able to recover and Zabbix's dashboard was updated to show "RECOVERED" as the status.

Table 2: SYN Flooding Test Results

No	Start Time of Attack (24hrs)	Alert Trigger	Problem listed on Zabbix Dashboard	Administrator's Alert (24hrs)	Time Attack is Stopped (24hrs)	Time Problem is Recovered (24hrs)
1	19:50	PROBLEM	10 packets without ACK from 218.111.111.246 !	19:51	19:54	19:52
2	20:10	PROBLEM	> 10 packets without ACK from 218.111.111.246 !	20:10	20:15	20:13
3	20:25	TRIGGER IS SILENT	Problem not listed	Alert not sent	21:35	-

During the SYN flood attack tests, Zabbix was able to detect an attack within 60 seconds of it being launched, and an alert was sent to the administrator immediately after it was detected. Zabbix was able to recover from these attacks between 2-5 minutes after they were launched, even before the attack was stopped. However, after the second attempt at testing Zabbix's capabilities to alert the administrator of the attack, Zabbix was not able to detect the attacks, and thus the administrator remained unaware of the attack. This gives attackers the chance to launch multiple SYN flooding attacks at the same time to use up the resources of the victim server.

Based on the results obtained during the testing, Zabbix was able to detect a possible ping flood and send an alert to the administrator within 60 seconds when the flooding was detected. For Zabbix to detect a possible ping flood, it has a minimum of a 30 second refresh time to fetch the data from the host monitored. Once the data from *tcpdump.log* is obtained, if the trigger requirement of more than 10 pings per second from the same source is met, Zabbix produces an alert for the dashboard and administrator. The recovery time has a recovery error of 60 seconds based on the time the attack was stopped and the time the recovery alert was sent to the administrator and for Zabbix to update its dashboard to show that the victim server has recovered from the attack. Figure 5 shows the message received by the administrator during the SYN flooding attack detected.



```
Problem: >10 packets without ACK from 218.111.111.246 !
Problem started at 20:10:41 on 2021.06.16
Problem name: >10 packets without ACK from 218.111.111.246 !
Host: Agent
Severity: Average
Operational data: Current: 74
Original problem ID: 980
8:10 PM
```

Figure 5: Alert Received by the Administrator in Telegram During SYN Flooding Attack

CONCLUSION AND RECOMMENDATIONS

This project was able to detect ping and SYN flooding attempts and send alerts to the administrator. A ping flood attempt of 10 or more pings from the same source per second was considered a ping flood attempt. Zabbix was able to detect the pings and log them as a possible ping flood attack. When the ping flood was detected, the Zabbix dashboard showed a spike in the ping flood graph, and a problem also appeared on the dashboard. The detection of an SYN flood was triggered by 10 or more SYN packets with no ACK, which was logged as an SYN flooding attack. When the attack was launched, the Zabbix dashboard showed the SYN flood as a problem. When both the flooding attacks were detected by Zabbix, a Telegram message with the details of the attack was sent to the administrator. Therefore, the objectives of the project were achieved. During the testing phase, several weaknesses in Zabbix's monitoring were found. The most notable finding is that Zabbix was not able to detect SYN flooding attacks as accurately as ping flood attacks. Zabbix was also able to solve a SYN flood attack within minutes of it being launched without having the attack stop. From this observation, it was concluded that Zabbix has a cut-off threshold for the number of SYN packets it receives. Zabbix also needs a period of at least 30 seconds to collect the data from the agent.

Future work can be done to include more types of attacks, particularly DDoS attacks. This project only used ping flooding and SYN flooding attacks. Ping of Death, SSH brute force, UDP floods, and Smurf attacks are just some of the attacks that could be used in future work. Adding attacks for more extensive monitoring on a server is a good investment as cybercriminals are always ready to exploit any weakness they find for their gain. SYN flooding detection needs further refinement as Zabbix was not able to accurately detect all the SYN floods conducted against it after more than 2 attacks. Next, it is also recommended that countermeasures be put in place after detecting the attacks. The countermeasures can include blocking the IP launching the attack or closing certain ports.

There are also several limitations to this project. The triggers used to identify a ping or SYN flooding attack may not reflect real-world situations as the volume of these attacks can be significantly larger. The SYN flooding detection was only able to pick up on two SYN flooding attempts during the testing that was done.

ACKNOWLEDGEMENT

The authors appreciate the reviewers for their contributions towards improving the quality of this research.

CONFLICT OF INTEREST DISCLOSURE

All authors declare that they have no conflicts of interest to disclose.

REFERENCES

Abid, K. (2020). Ping Flood Attack Detection via Wireshark. *International Journal of Advanced Science and Technology*, *29*(5), 9595–9601.

Barbu, I. D., Pascariu, C., Bacivarov, I. C., Axinte, S. D., & Firoiu, M. (2017). Intruder monitoring system for local networks using python. *Proceedings of the 9th International Conference on Electronics, Computers and Artificial Intelligence*, 1–4. https://doi.org/10.1109/ECAI.2017.8166457.

Birkinshaw, C., Rouka, E., & Vassilakis, V. G. (2019). Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks. *Journal of Network and Computer Applications*, *136*(February), 71–85. https://doi.org/10.1016/j.jnca.2019.03.005.

Gayathri, R., & Neelanarayanan, V. (2018). DoS detection solution for cloud platform using SNMP. *International Journal of Pure and Applied Mathematics*, *118*(23), 175–183.

Hakim, A. R., Rinaldi, J., & Setiadji, M. Y. B. (2020). Design and Implementation of NIDS Notification System Using WhatsApp and Telegram. *8th International Conference on Information and Communication Technology*, 4–7. https://doi.org/10.1109/ICoICT49345.2020.9166228.

Johnson, R., & Elizabeth, N. E. (2018). Network's server monitoring and analysis using Nagios. *Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking*, 1904–1909. https://doi.org/10.1109/WiSPNET.2017.8300092.

Mardiyono, A., Sholihah, W., & Hakim, F. (2020). Mobile-based Network Monitoring System Using Zabbix and Telegram. *International Conference on Computer and Informatics Engineering*, 473–477. https://doi.org/10.1109/ic2ie50715.2020.9274582.

Nobles, C. (2018). Botching Human Factors in Cybersecurity in Business Organizations. *HOLISTICA – Journal of Business and Public Administration*, *9*(3), 71–88. https://doi.org/10.2478/hjbpa-2018-0024.

Salunkhe, H. S., Jadhav, S., & Bhosale, V. (2017). Analysis and Review of TCP SYN Flood Attack on Network with Its Detection and Performance Metrics. *International Journal of Engineering Research And*, *V6*(01), 250–256. https://doi.org/10.17577/ijertv6is010218.

Sulistya, I. M. A., & Sasmita, G. M. A. (2020). Network Security Monitoring System on Snort with Bot Telegram as a Notification. *International Journal of Computer Applications Technology and Research*, 9(2), 059–064. https://doi.org/10.7753/ijcatr0902.1004.