



## Design of network monitoring system based on LibreNMS using Line Notify, Telegram, and Email notification



Nurwan Reza Fachrurrozi<sup>1\*</sup>, Andri Agustav Wirabudi<sup>1,2</sup>, Seandy Arandiant Rozano<sup>1</sup>

<sup>1</sup>Department of Electrical Engineering, Faculty of Engineering, Institut Teknologi Telkom Jakarta, Indonesia

<sup>2</sup>Department of Intelligent Media Engineering, Hanbat National University, South Korea

### Abstract

*Institut Teknologi Telkom Jakarta (IT Telkom Jakarta) is an educational institution that supports student activities and provides internet capabilities to implement online learning systems. As the number of students increases with every year, so does the use of the internet and intranet networks and the experienced network problems. A network administrator is a person who is responsible for managing a computer network. Network administrators usually face network problems in monitoring network devices. This is because the process and operation are done manually. This means network administrators need direct access to the location to monitor all resources. Therefore, a network device monitoring system is needed to manage network devices centrally. This research focuses on the problem of monitoring network devices using open-source tools and software. Based on the implementation results, free network monitoring software such as LibreNMS can track and monitor all devices in all conditions and notify the active device condition in case of network failure such as up, down, reboot to the administrator via Line Notify, Telegram, and Email. With this network monitoring system, IT Telkom Jakarta is expected to be able to implement an integrated and well-monitored internet network system. Besides, the results of this study also produce real-time data on bandwidth usage, logging problems, and resource availability. This can significantly improve network availability and security.*

### Keywords:

Email;  
LibreNMS;  
Line Notify;  
Network Monitoring System;  
Telegram;  
VirtualBox;

### Article History:

Received: July 18, 2022

Revised: October 8, 2022

Accepted: October 30, 2022

Published: February 2, 2023

### Corresponding Author:

Nurwan Reza Fachrurrozi  
Department of Electrical  
Engineering, Faculty of  
Engineering, IT Telkom Jakarta,  
Indonesia  
Email: [nurwan@ittelkom-jkt.ac.id](mailto:nurwan@ittelkom-jkt.ac.id)

Copyright ©2023 Universitas Mercu Buana

This is an open access article under the [CC BY-NC](https://creativecommons.org/licenses/by-nc/4.0/) license



### INTRODUCTION

Network monitoring is consistent monitoring of computer network disruptions and breakdowns to ensure continued network performance. Managing a complex network has three considerations: network structure, management, and effectiveness [1]. NMS needs to accurately track information about network devices such as servers, routers, switches, and endpoint devices that are running unchanged. If a slow or faulty device is detected, the network monitoring software will notify the network administrator about the problem. Network failures often go unnoticed by network administrators.

This can lead to severe network degradation issues through time-consuming troubleshooting. The problem occurs when the system operator is unable to monitor the applications running on the Network Operations Center NMS. This may be due to personal or additional activities requiring the system operator to check the server room's network operation.

In [2], the procedure used in this study is the observation by looking at and observing the server and network at the Limited liability company Indonesian Railways DAOP 3 directly. In [3], a cloud monitoring system based on an open-source was designed and implemented by

subdividing it into CPU, memory, storage, and network parts. In [4], this study aims to monitor the proxy wireless local area network (WLAN) using the Dude application at Musamus University. In research [5], this study to Development of a Prototype of Open-Source Network Management System.

This research paper describes research to develop a prototype of an open-source network management system to provide a free solution to network monitoring problems at IT Telkom Jakarta.

## MATERIAL AND METHODS

### Network Monitoring Tools

IT Telkom Jakarta needs a responsive network monitoring tool to help network administrators easily detect and troubleshoot information asset failures. If a resource such as a switch is turned off, the problem is that the user cannot access the network. Delayed initial troubleshooting while analyzing root causes of devices with specific issues affecting service. A related application that helps minimize the impact of the problem is LibreNMS [6].

LibreNMS has many features such as reporting, event handling, and resource monitoring (CPU load, memory, up/down status, uptime, traffic, bandwidth). One of the outstanding features of LibreNMS is the explosion notification alert. This is a feature that works if the network has a problem with one of the devices. This feature notifies the network administrator or authorized person in a particular department about network errors. Device failures can be categorized based on the parameters set or created by the network operator. Error notifications can be sent to various means of communication via Line Notification Bots, Telegram, and Email.

This study used experimental methods. This means a kind of research aimed at finding some degree of solidity in the symptoms of the event. This allows us to undoubtedly establish previously unknown causes and consequences. Thus, this study has the property of revealing causal factors that are a comparison of pre-and post-experimental conditions. The network development methodology used in this research is NDLC, a step-by-step or step-by-step approach to analyze and create designs using specific user activity cycles. NDLC has several phases.

The stages are A. Data collection (analysis) This phase is the first phase in developing computer networks. During this phase, two activities were carried out to analyze the existing network topology and collect data for

all devices used by IT Telkom Jakarta. The two activities are the first observation, namely observing the object directly and obtaining information that can be used as a reference when designing the system. 2. Direct interview from IT Telkom Jakarta to collect information about the research topic by asking questions directly. B. Design After completing the analysis phase, the proposed network topology design is generated. We hope that the proposed network topology provides an overview of the existing requirements at IT Telkom Jakarta. The integration between the Email Server, Telegram Server, and Administrator components is shown in [Figure 1](#) and [Figure 2](#) [7].

The research method uses PPDIOO which Cisco developed in designing network systems. The stages in the PPDIOO method include Prepare, Plan, Design, Implement, Operate and Optimize, which are the methods used by Cisco in network design. So that in this study it is used in the design of network monitoring using Line Notify, Telegram & Email notifications. The stages carried out in this research method can be shown in [Figure 1](#).

What is the NDLC's purpose in the paper, please elaborate. The network development methodology used in this research is NDLC, a step-by-step or step-by-step approach to analyze and create designs using specific user activity cycles. NDLC has several phases. The first stages is Data collection (analysis) This phase is the first phase in the development of computer networks. During this phase, two activities were carried out to analyze the existing network topology and collect data for all devices used by IT Telkom Jakarta. The two activities are the first observation, namely observing the object directly and obtaining information that can be used as a reference when designing the system. Direct interview from IT Telkom Jakarta to collect information about the research topic by asking questions directly. Design After completing the analysis phase, the proposed network topology design is generated. We hope that the proposed network topology provides an overview of the existing requirements at IT Telkom Jakarta. The integration between the Email Server, Telegram Server, and Administrator components is shown in [Figure 1](#) and [Figure 2](#).

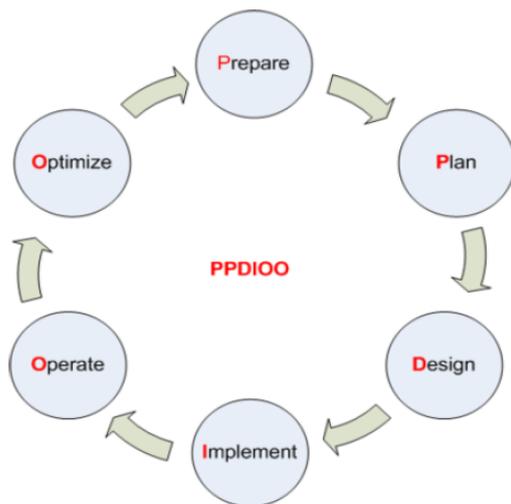


Figure 1. PPDIIO [8]

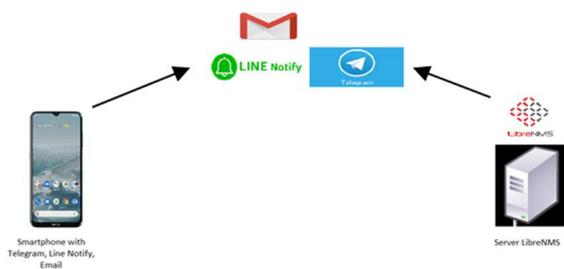


Figure 2. Telegram, Line Notify, Email Architecture [9][10].

Prepare is a problem analysis that performs a literature search on Line Notify Bot, Telegram Bot API, and Network Notification Email to identify problems and systems needed to solve problems [11, 12, 13]. The plan is the stage of system design and requirements analysis, whether it meets the required requirements. Design is the stage of making a system to build. From topology to system configuration design. An implement is the network monitoring phase, Line Notify Telegram bot notifications, and Email applications. Operate is an experiment to execute a bot program that is created. Optimize performs optimization on all design components. Start with other bot design requirements, such as monitoring hardware, software, and network impact, and add the necessary components.

The procedure for using the PPDIIO research level goes through optimization. Preparation, planning, design, implementation, operation, and optimization. This research is indirectly useful for administrators to monitor their networks, especially with IT Telkom Jakarta. According to research, one of the recommended tools for monitoring is LibreNMS. Given that IT Telkom Jakarta did not implement the LibreNMS

tools, this research activity is about configuring LibreNMS to run on the IT Telkom Jakarta networks. Can be used for network monitoring processes.

In addition, proper configuration of devices, sensors, and connections must be designed and implemented to meet network monitoring requirements. LibreNMS receives relevant data and parameters for network monitoring purposes in IT Telkom Jakarta. Then, after getting some data, how to interpret the data so that can view the performance of IT Telkom Jakarta's previous computer networks.

The survey was conducted by the IT Telkom Jakarta DTI (Information Engineering Support) Unit, the unit responsible for providing information and communication technology services to all IT Telkom Jakarta. Specifically, network monitoring is performed on the IT Telkom Jakarta campus networks system in the "Board / Staff / Sysfo" building. This letter has three purposes. The first is the development of a LibreNMS. Second, for the second purpose, is this system should be able to monitor network resources on the networks. Finally, the main purpose is to inform and identify resource issues for the administrator.

### Research Design

A monitoring system used by the Mikrotik Routerboard & Ruijie that sends notifications via the Line Notify, Telegram & Email applications. The parameters used for this survey are mobile responses and UP and DOWN notifications. The study was conducted by designing a remote monitoring system using an existing architecture, as shown in Figure 3.

### Network Architecture

The network architecture describes the topology and overview of the network system used. The monitoring system runs on the LAN network. Used LibreNMS Notification API surveillance system and sent notifications to Line Notify, Telegram, and Email applications. After conducting the analysis, the researcher's next stage was the design stage. The design or network topology installed at IT Telkom Jakarta was obtained after conducting observations and interviews.

### System Components

The implementation process has hardware and software components, as described in Table 1. The hardware used in this study is one of the ACER ASPIRE A5153 notebooks. Table 2 shows the software used in this study.

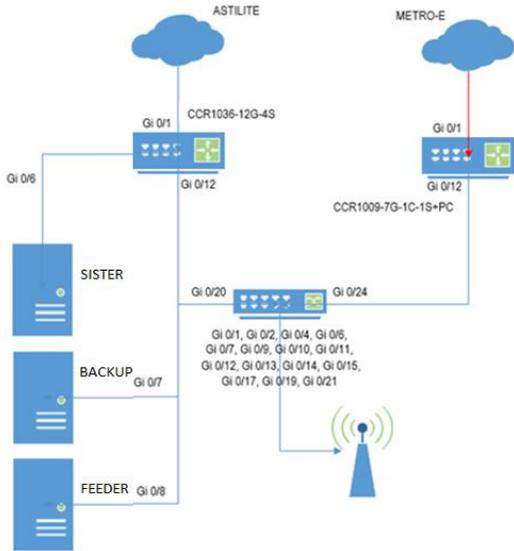


Figure 3. Topology Network

Table 1. Hardware [14]

No	Hardware
1	Processor Intel Core i3 1005G1
2	SSD 512 GB / RAM 20 GB
3	OS Microsoft Windows 10 Home Edition Single Language 2020

Table 2. Software [15]

No	Software	Remarks
1	Ubuntu Server 20.04	Operating system for running the system
2	LibreNMS and its supporting software	Monitoring software to implement
3	MySQL	As a database server
4	Apache	As Web Server for LibreNMS Interface
5	Net-SNMP	As a tool to run SNMP

The research was conducted at DTI (Infrastructure Technical Support) IT Telkom Jakarta, the center for information and communication technology services for all IT Telkom Jakarta. This research is supported and financed in an independent grant research program by LPPM-IT Telkom Jakarta (Institute for Research & Community Development - IT Telkom Jakarta).

This research was conducted in the existing IT Telkom Jakarta network, where the benefits are useful in optimizing the monitoring of the IT Telkom Jakarta network in maintaining stability and network performance. The investigation process begins with the installation of the PuTTY application for remote access to the server and other necessary software such as MySQL and Apache. After installing and configuring LibreNMS, the next step is to make sure LibreNMS is running properly. If not, make sure the configuration process is complete. After successful installation and configuration, the next

step is to add the IP of the monitored device to LibreNMS [16]. After successfully adding devices to the LibreNMS application, we can then proceed to the next phase of using LibreNMS to test and measure network traffic and resources. The data used is incoming and outgoing traffic and resource usage on each network device, such as the temperature at which this data was obtained through memory, CPU usage, and the results of LibreNMS network monitoring. The analysis results obtained from this study will be a quick reference for network administrators to determine whether the A campus network at Daan Mogot IT Telkom Jakarta is still functioning properly or needs improvement. Figure 4 is a flow chart for designing a monitoring system using LibreNMS. Monitoring resource utilization in the form of memory, CPU usage, and temperature for all monitored devices. LibreNMS monitoring results are presented in a percentage format for easy reading by network administrators. Traffic monitoring allows users to see which ports are in use and which are not. Users can also see the total bandwidth usage, both inbound and outbound, in the form of real-time, average, and maximum data usage. LibreNMS monitoring of the total memory & storage usage of IT Telkom Jakarta network switches. Administrators can use this monitoring data to decide whether to increase the device's storage capacity. LibreNMS monitors total CPU usage on the IT Telkom Jakarta switch network. It can be seen that the CPU utilization on the device under normal conditions, the processor utilization is indicated by the CPU utilization which does not exceed 80% of the processor utilization.

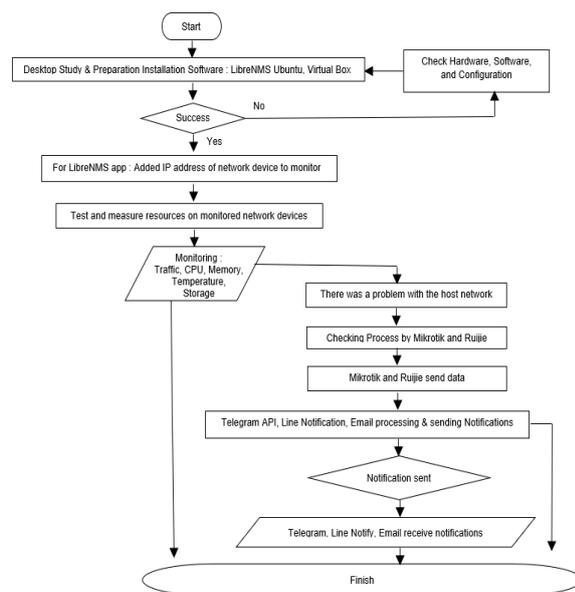


Figure 4. Research Flowchart

LibreNMS temperature monitoring generated by all IT Telkom Jakarta switch devices. The temperature monitor shows the number of temperatures recorded by each monitor. This helps keep the temperature generated by the device normal and can be used as a reference for administrators to decide whether to take further action.

- LibreNMS functions for monitoring Traffic, CPU, Memory, Temperature, Storage, dBm, State, and Fan Speed.
- It would be beneficial to specify the failure function standard for these items.

Monitoring resource utilization in the form of memory, CPU usage, and temperature for all monitored devices. LibreNMS monitoring results are presented in a percentage format for easy reading by network administrators. Traffic monitoring allows users to see which ports are in use and which are not. Users can also see the total bandwidth usage, both inbound and outbound, in the form of real-time, average, and maximum data usage. LibreNMS monitoring of the total memory & storage usage of Telkom Jakarta IT network switches. Administrators can use this monitoring data to decide whether to increase device storage capacity.

LibreNMS monitoring total CPU utilization in the switch network of Sudirman Campus, Udayana University. Looking at CPU utilization, we can see that under normal conditions, processor utilization is characterized by CPU utilization that does not exceed 80% of processor utilization.

LibreNMS temperature monitoring generated by all IT Telkom Jakarta switch devices. The temperature monitor shows the number of temperatures recorded by each monitor. This helps keep the temperature generated by the device normal and can be used as a reference for administrators to decide whether to take further action.

## RESULTS AND DISCUSSION

The built system uses the Ruijie monitoring system, and the Mikrotik router uses LibreNMS notifications to test the network connectivity of each host. When the status changes on a particular interface, the system sends Line Notifications, Telegrams, or Email notifications to the administrator's smartphone. In this study, network monitoring uses Telegram Bot API notifications, the Line Notify Bot API and automatically executed email [17][18].

RouterOS Mikrotik, Ruijie's monitoring system, integrated with Telegram's API token LineNotify, automatically tests connections via email if the connection is lost or no connection.

The monitoring system sends a notification from the router to the Telegram application Line Notify via the previously built-in token address to the email client. This application can be used to monitor the results of monitoring measurements in the form of network monitoring data. The application can display the traffic in the form of notification message Telegrams, Line Notifications, and Emails received by the administrator. The step installation LibreNMS & Active Device Configuration: First, install Ubuntu Server version 20.04 and activate remote access via the PuTTY software. After that, prepare the MySQL / Apache configuration. Then configure IP address, hostname [19][20]. Figure 5 shows the LibreNMS Add Device menu.

Once all the devices have been added, the watched person can see the status of the network devices in the LibreNMS dashboard. Figure 6 shows that the device is being monitored. In addition, users can view many device parameters such as IP address, port number, device type, operating system used, and power on/off time. This is the result of the status of the monitored device. On the display, one of the IPs connected to Mikrotik will be monitored in Figure 10.

In the "Availability Map" view of the total hosts up, down, and warn in Figure 7. In the All Devices view, monitoring several connected devices is shown in Figure 8.

The image shows the 'Add Device' configuration page in LibreNMS. It features a form with the following elements:
 

- Hostname or IP:** A text input field.
- SNMP:** A dropdown menu set to 'ON'.
- SNMP Version:** A dropdown menu set to 'v2c'.
- Port Association Mode:** A dropdown menu set to 'Index'.
- Community:** A text input field.
- Force add:** A checkbox set to 'OFF' with the note '(No SNMP or SNMP checks performed)'.
- Buttons:** 'Add Device' and 'Cancel' buttons.

Figure 5. Add Device View

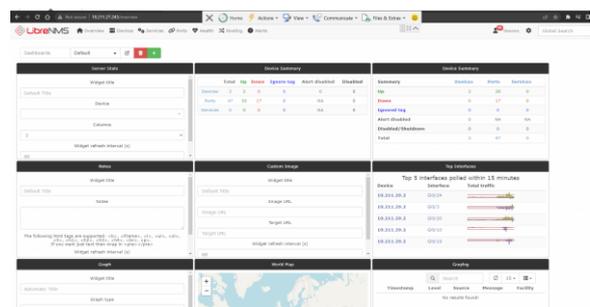


Figure 6. DASHBOARD LibreNMS IT Telkom Jakarta

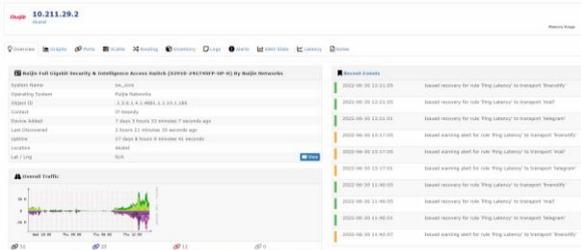


Figure 7. Availability Map Only Device on LibreNMS



Figure 8. All Devices on LibreNMS



Figure 9. Device Ruijie [21]

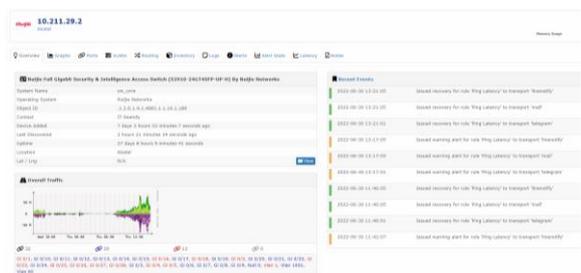


Figure 10. Device Mikrotik [22][23].

On the display, one of the IPs connected to Ruijie will be monitored in Figure 9.

### Notify by Telegram, Line notification, Email, Line Notify

In the notification, first signs up for the @BotFather account. Bot Father, <https://notify-bot.line.me/my/>, Email-Google is the official Line Notification, Telegram, email responsible for creating new bots. The bot provides a token to access the bot by entering the name and bot ID that will be created after entering it. Similarly, enter a password in the libreNMS notification email account to use the Google account email as the application's account. Figure 11 shows the initial step of making this Telegram bot is to have a telegram account itself, then to create a Telegram bot account the author looks for a Telegram id named @BotFather and makes a Telegram bot [24].

Figure 12 shows the creation of this Telegram group aimed at the notification of network device problems for all network technicians who work.

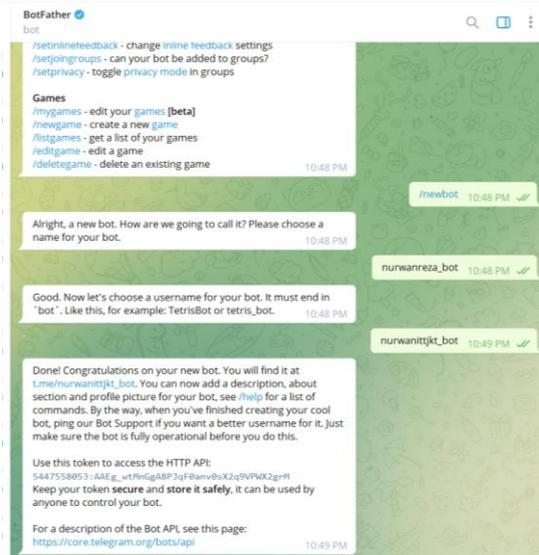


Figure 11. Telegram API Token

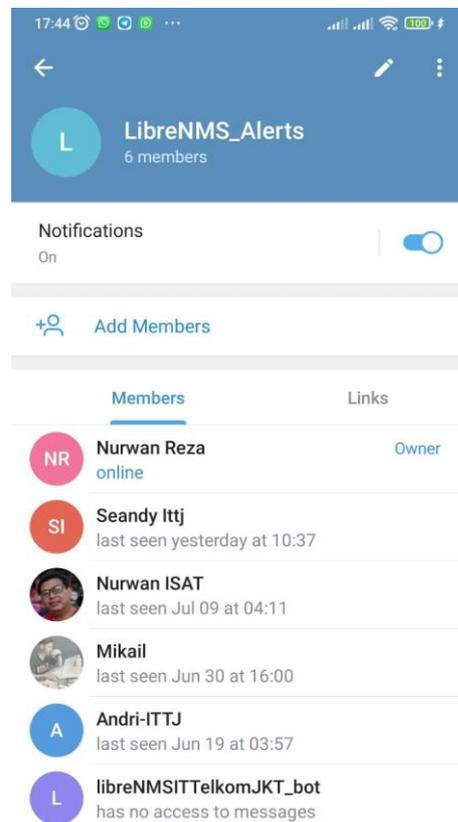


Figure 12. Telegram Groups

Figure 13 shows the initial step of making this notify line bot is to have a line notify account itself, then to create a line notify bot account the author looks for the line notify id configured at <https://notify-bot.line.me/my/> and makes the bot line notify. Figure 14 shows the creation of this line notify group which aims to notify network

device problems not only for 1 network technician but for all network technicians who work

### Setup Email transport in LibreNMS

If use Gmail-related email addresses, enable less secure apps. Figure 15 shows the steps for setting up email to allow insecure apps to access the account.

### Configure LibreNMS Notifications

The transport is located at LibreNMS / Alert / Transport / and can be configured in Alerts-> Alert Transport in the Web UI. Contacts are automatically collected and forwarded to the configured transport.

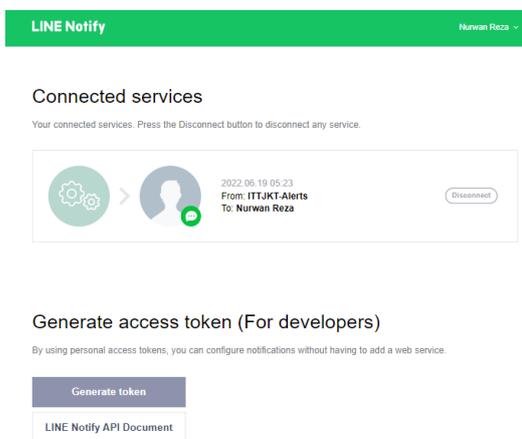


Figure 13. Line Notify API Token



Figure 14. Line Notify Groups

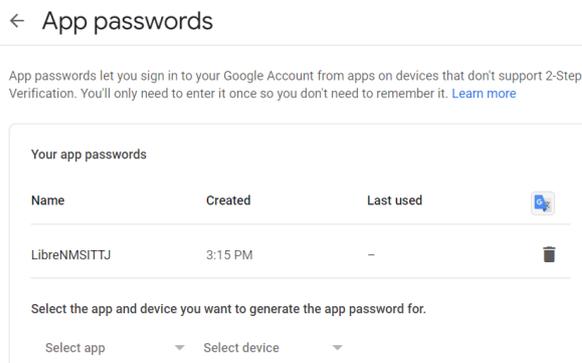


Figure 15. Password Account Email

By default, contacts are collected only when an alert is triggered and future changes to the incident's contacts are ignored. If want to regain contacts each time send them, set Update Ignored Contact Email Addresses to Off in the Web UI. Contacts always include the LibreNMS user who has at least read permission for the entity sending the alert, in addition to the SysContact defined in the device's SNMP configuration. At this time, LibreNMS only supports port or device permissions. Can exclude SysContact by switching "Issue warning to SysContact". Need to toggle the option to include users with global read, admin, or normal user rights.

- Issue a warning to the administrator.
- Issue a warning to read-only users
- Warn normal users.

Figure 16 When a specific event occurs (eg: Link Down) NMS should be able to send administrators some notifications. On LibreNMS this is done by creating different Transports and Alerting Rules. Transports are the mechanism on how LibreNMS should send the alerts (maybe by Email, Telegram, Line Notify), where Alerting Rules specify when and to whom to send those alerts.

### Setup Email transport in LibreNMS

Follow the steps to configure emails in Figure 17 and Figure 18.

Transport Name	Transport Type	Default	Details
Telegram	Telegram	Yes	Chat ID: -661355916 Token: 5451791690-AAE9gTYqP2WXX_wmAD-iMe6A3_030hW5Xk Format: HTML
Admin Email	Mail	Yes	Email: nurwan@ittelkom-jkt.ac.id
ITTJKT_Line Notify LibreNMS_Al	LINE Notify	Yes	Token: 2M5f6v9gBQhW5vEujRjE5zQh4DvYRoshUvDvXDM

Figure 16. Configure Alert Transport in LibreNMS

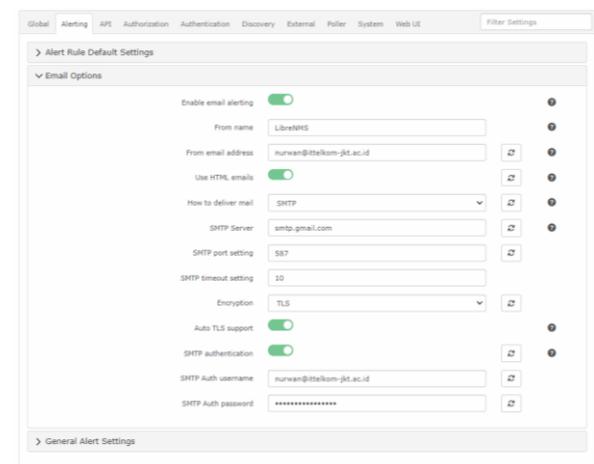


Figure 17. Configure the Email Option in LibreNMS

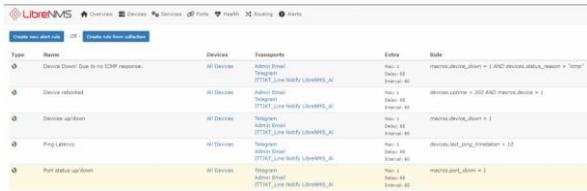


Figure 18. Configure Email Alert Rules in LibreNMS

**Monitoring**

LibreNMS Data Usage Monitoring – LibreNMS Data Traffic Monitor can monitor inbound and outbound traffic. The result of inbound and outbound traffic values on one of the switches on the IT Telkom Jakarta campus: the Mikrotik Router and the Ruijie Switch. Using these results, the administrator can see how traffic is being used on each switch through the LibreNMS application.

Figures 19 and Figure 20 show traffic diagrams for the Ruijie Switch & Mikrotik Router. In the graph, it can be seen that there is a vertical line that shows the bandwidth in megabits/second and a horizontal line that shows the observation period. There is also an up-graph of incoming traffic and a down-charting of outgoing traffic. Traffic monitoring results allow users to see which ports are in use and which are not. Users can also view total bandwidth usage,

both inbound and outbound, in real-time, average, or maximum data usage.

**Monitor Resource Usage**

This section displays resource usage monitoring results in terms of memory, CPU utilization, and temperature for all monitored devices that have resource usage monitoring results. Figure 21 shows the results of LibreNMS monitoring of total storage usage in the IT Telkom Jakarta Switched Campus Network. Monitoring result data is displayed as a percentage of the administrator's readability. Administrators can use this monitoring data to determine if the device needs more storage space.

Users can also view multiple device columns, storage fields, usage fields, and usage fields while monitoring storage usage. For memory usage, users can display multiple colors in a column. The color indicates memory usage. Figure 22 shows the total CPU utilization on the network. Monitoring result data is displayed as a percentage of the administrator's readability. Administrators can use this monitoring data to determine if CPU utilization is still normal. CPU utilization is indicated by CPU utilization not exceeding 80% [25].

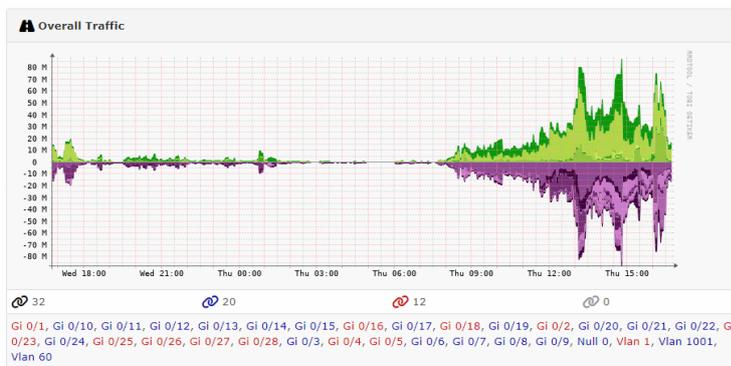


Figure 19. Ruijie Traffic

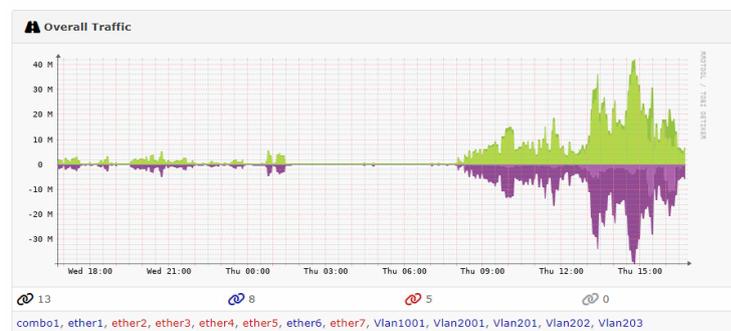


Figure 20. Mikrotik Traffic

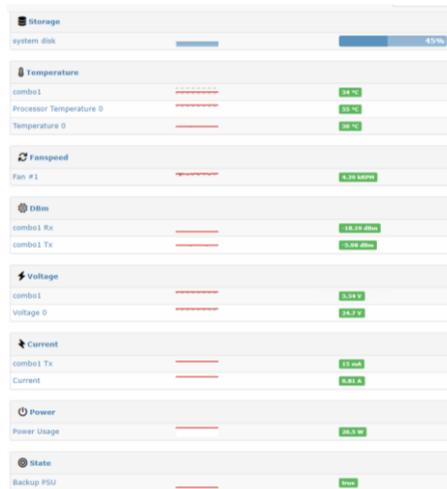


Figure 21. Health Storage, Temperature, Fan speed, dBm, Voltage, Current, Power, State

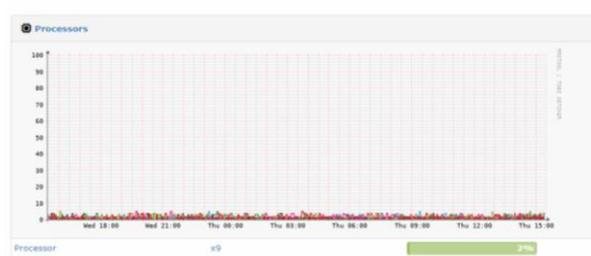


Figure 22. Health Processor System

Like Temperature, monitoring data shows the number of temperatures recorded by each sensor. Temperature monitoring allows users to view multiple device columns, sensor fields, current columns, and bottom and top columns.

Figure 23 shows LibreNMS monitoring the memory usage of a Mikrotik router. Monitoring result data is displayed in percentage format for administrator readability. Administrators can use this monitoring data to decide whether to increase device storage capacity. While monitoring the storage usage, the user can also see several columns of information such as B.: Device column, Storage column, Used column, and Usage column. For memory usage, users can see some colors in the Used column. This color indicates memory usage

Figure 23 shows LibreNMS monitoring the memory usage of a Mikrotik router. Monitoring result data is displayed in percentage format for administrator readability. Administrators can use this monitoring data to decide whether to increase device storage capacity. While monitoring the storage usage, the user can also see several columns of information such as: B.: Device column, Storage column, Used column, Usage column. For memory usage, users can see some colors in the Used column. This color indicates memory usage.



Figure 23. Health Memory System

### Notification

The ultimate goal of this investigation is to alert administrators to device issues and find them. When the device port moves up or down, the administrator receives the following notifications: The notification also includes the time stamp of the up / down port on the device. There is also a ping delay failure. In Figures 24, Figure 25, and Figure 26 the test focuses on the functional requirements to see if the system produces the desired output and is under the function. Testing can be done by disabling or disabling the interface on the Mikrotik router, the Ruijie Switch, and also the real-time of the installed device. Another way is to unplug the LAN cable on the connected device.



Figure 24. Telegram Alert: Port Status UP / Down

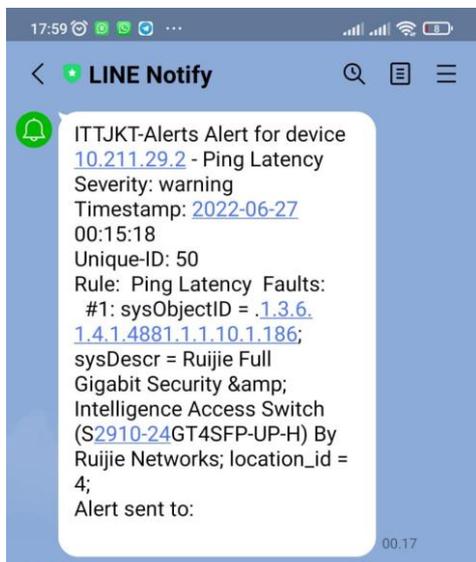


Figure 25. Line Notify Alert: Ping Latency Device



Figure 26. Email Notification: Ping Latency Device

These two methods are carried out so that the system detects that the network device is in a down condition and sends notification messages of disturbances via Bot Line Notify, Telegram, and Email. The figure above shows that when the switch & router or with the device name 10.211.29.2 is up and down librenms managed to send information on Telegram, Email, and Line Notify. Please investigate quantitatively the alert

notification performance for each failure scenario (successful percentage).

## CONCLUSION

Has successfully installed LibreNMS which is used for network monitoring purposes in the IT Telkom Jakarta environment. LibreNMS monitoring system has been able to display traffic, memory, CPU usage, and temperature. Resource monitoring shows that the device memory is normal and the memory has not reached 100%, and the device CPU is normal and does not exceed the threshold. Optimization of network monitoring using the libreNMS feature based on Line Notify, Telegram, and Email for notification of network problems, able to assist in creating a network monitoring system that can make it easier for administrators to unite network devices so that devices remain active and operate according to their duties, to quickly make repairs to devise problems dead network. In optimizing network monitoring using libreNMS, the Mikrotik Router and Ruijie Switch must be connected to the internet 24 hours a so that when Mikrotik, Ruijie is not connected to the internet there is no notification of network problems on Line Notifications, Telegram and Email. To optimize network monitoring with libreNMS, the mobile administrator must be connected to the internet. If the mobile administrator is not connected to the internet, the Notify line, Telegram, and email will not notify about network problems.

## ACKNOWLEDGMENT

This research was supported and financed by the Independent Grant Research Program by LPPM-IT Telkom Jakarta (Institute for Research & Community Development - IT Telkom Jakarta). We would like to thank colleagues at IT Telkom Jakarta who have provided insight and expertise that greatly assisted this research.

## REFERENCES

- [1] Z. Aouini and A. Pekar, "NFStream," *Computer Networks*, vol. 204, ID: 108719, February 2022, doi: 10.1016/j.comnet.2021.108719.
- [2] K. Upreti, V. Kumar, D. Pal, M. S. Alam, and A. K. Sharma, "Design and Development of Tracking System in Communication for Wireless Networking," *Lecture notes in networks and systems*, Dec. 2021, doi: 10.1007/978-981-16-6369-7\_19.
- [3] E. Jo and H. Yoo, "Implementation of Cloud Monitoring System Based on Open Source Monitoring Solution," *Software Engineering in IoT, Big Data, Cloud and Mobile*

- Computing. Studies in Computational Intelligence*, vol 930, Springer, Cham, Jan. 2021, doi: 10.1007/978-3-030-64773-5\_15.
- [4] F. X. Manggau, A. Latif, and Suwarjono, "E-Monitoring Microtic Network uses the Dude in Musamus University," in *Journal of Physics: Conference Series*, vol. 1569, no. 2, July 2020, doi: 10.1088/1742-6596/1569/2/022024.
- [5] Salihah Zakaria, A. Nasir, Syahrul Fahmy, and Nooraida Samsudin, "Development of a Prototype of Open-Source Network Management System," *International Journal of Synergy in Engineering and Technology (IJSET)*, vol. 1, no. 2, pp. 38-45, Dec. 2020.
- [6] M. Laštovička, M. Husák and L. Sadlek, "Network Monitoring and Enumerating Vulnerabilities in Large Heterogeneous Networks," *NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, 2020, pp. 1-6, doi: 10.1109/NOMS47738.2020.9110394.
- [7] M. Rath, B. Pati, C. R. Panigrahi, and J. L. Sarkar, "QTM: A QoS Task Monitoring System for Mobile Ad hoc Networks," *Advances in intelligent systems and computing*, vol. 707, Jan. 2019, doi: 10.1007/978-981-10-8639-7\_57.
- [8] Y. Li, S. Xia, M. Zheng, B. Cao and Q. Liu, "Lyapunov Optimization-Based Trade-Off Policy for Mobile Cloud Offloading in Heterogeneous Wireless Networks," in *IEEE Transactions on Cloud Computing*, vol. 10, no. 1, pp. 491-505, 1 Jan.-March 2022, doi: 10.1109/TCC.2019.2938504.
- [9] E. A. Z. Hamidi, L. S. Dzudin, A. Faroqi, and M. A. Ramdhani, "The Implementation of Alert System for LAN Network Monitoring Using the Dude Based Email," in *IOP Conference Series: Materials Science and Engineering*, vol. 288, no. 1, Jan. 2018, doi: 10.1088/1757-899X/288/1/012054.
- [10] M. H. Ibnu Hajar, A. W. Dani, and S. Miharno, "Monitoring of Electrical System Using Internet of Things with Current Electric Sensors," *SINERGI*, vol. 22, no. 3, pp. 211-218, Oct. 2018, doi: 10.22441/sinergi.2018.3.010.
- [11] F. W. Christanto and M. Sani Suprayogi, "Enhancement Network Monitoring System Functionality by Implementing an Android-based Notification System to Monitor Virtual Servers on Cloud Computing Network," *International Journal of Applied Information Technology (IJAIT)*, vol. 2, no. 01, pp. 7-15, May 2018, doi: 10.25124/ijait.v2i01.1002.
- [12] M. A. Rosid, A. Rachmadany, M. T. Multazam, A. B. D. Nandiyanto, A. G. Abdullah, and I. Widiaty, "Integration Telegram Bot on E-Complaint Applications in College," in *IOP Conference Series: Materials Science and Engineering*, vol. 288, no. 1, Jan. 2018, doi: 10.1088/1757-899X/288/1/012159.
- [13] P. Thoman *et al.*, "A taxonomy of task-based parallel programming technologies for high-performance computing," *Journal of Supercomputing*, vol. 74, no. 4, pp. 1422–1434, Apr. 2018, doi: 10.1007/s11227-018-2238-4.
- [14] A. H. Kobeissi, G. Lanza, R. Berta, F. Bellotti, and A. de Gloria, "Development of a Hardware/Software System for Proprioception Exergaming," *International Journal of Serious Games*, vol. 5, no. 2, pp. 87–100, Jun. 2018, doi: 10.17083/ijsg.v5i2.244.
- [15] M. R. Yaswinski, M. M. Chowdhury and M. Jochen, "Linux Security: A Survey," *2019 IEEE International Conference on Electro Information Technology (EIT)*, Brookings, SD, USA, 2019, pp. 357-362, doi: 10.1109/EIT.2019.8834112.
- [16] Á. L. V. Caraguay and L. J. G. Villalba, "Monitoring and discovery for self-organized network management in virtualized and software defined networks," *Sensors (Switzerland)*, vol. 17, no. 4, Apr. 2017, doi: 10.3390/s17040731.
- [17] C. Lebeuf, M. A. Storey, and A. Zagalsky, "Software Bots," *IEEE Software*, vol. 35, no. 1, pp. 18–23, Jan. 2017, doi: 10.1109/MS.2017.4541027.
- [18] I. Beschastnikh, M. F. Lungu and Y. Zhuang, "Accelerating Software Engineering Research Adoption with Analysis Bots," *2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Technologies Results Track (ICSE-NIER)*, Buenos Aires, Argentina, 2017, pp. 35-38, doi: 10.1109/ICSE-NIER.2017.17.
- [19] I. Mavridis and H. Karatza, "Performance evaluation of cloud-based log file analysis with Apache Hadoop and Apache Spark," *Journal of Systems and Software*, vol. 125, pp. 133–151, Mar. 2017, doi: 10.1016/j.jss.2016.11.037.
- [20] S. R. Majewski *et al.*, "The Apache Point Observatory Galactic Evolution Experiment (APOGEE)," *The Astronomical Journal*, vol. 154, no. 3, p. 94, Aug. 2017, doi: 10.3847/1538-3881/aa784d.

- [21] R. Huggins and D. Prokop, "Network structure and regional innovation: A study of university–industry ties," *Urban Studies*, vol. 54, no. 4, Mar. 2017, doi: 10.1177/0042098016630521.
- [22] D. Lacković and M. Tomić, "Performance analysis of virtualized VPN endpoints," *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, 2017, pp. 466-471, doi: 10.23919/MIPRO.2017.7973470.
- [23] A. Pratama, Agussalim and F. Muhammad Reza, "Point-to-Multipoint Implementation on Mikro Tik Router-Based Warehouses Networks," *2020 6th Information Technology International Seminar (ITIS)*, Surabaya, Indonesia, 2020, pp. 87-91, doi: 10.1109/ITIS50118.2020.9321045.
- [24] Y. Noguchi and F. Tanaka, "A pilot study investigating self-disclosure by elderly participants in agent-mediated communication," *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, Lisbon, Portugal, 2017, pp. 29-34, doi: 10.1109/ROMAN.2017.8172276.
- [25] S. Chaikhamwang, C. Jantahajakowit and S. Fongmanee, "IoT for Smart Farm: A Case Study of the Fertilizer Mixer Prototype," *2021 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering*, Cha-am, Thailand, 2021, pp. 136-139, doi: 10.1109/ECTIDAMTNCN51128.2021.9425708.