

## Implementasi Sistem *Monitoring* Jaringan Berbasis *Simple Network Management Protocol* Dengan Notifikasi Telegram

Ariella Giovanni Wahyuriyanto<sup>1\*</sup>, Heribertus Yulianton<sup>2</sup>

Teknik Informatika, Universitas Stikubank Semarang, Semarang, Indonesia

\*e-mail *Corresponding Author*: aariella22@gmail.com

### Abstract

*The Faculty of Science and Mathematics at Diponegoro University faces challenges managing information technology devices through the Unit Pengelolaan dan Pelayanan Teknologi Informasi (UP2TI). This problem has an impact on the efficiency and effectiveness of network management. This study aims to evaluate the implementation of LibreNMS as a network management system that can improve the efficiency of management by the UP2TI team. The method used is Network Development Life Cycle, which analyzes features such as Alert Notification and Validate Config. In addition, the Network Monitoring System, based on a Simple Network Management Protocol (SNMP) with telegram notifications, has been implemented to facilitate technicians in network monitoring. The results showed that using LibreNMS and telegram notification systems increases the efficiency of monitoring and managing network devices.*

**Keywords:** *Simple Network Management Protocol; LibreNMS; Network Development Life Cycle; Simple Network Management Protocol*

### Abstrak

Fakultas Sains dan Matematika Universitas Diponegoro menghadapi tantangan dalam mengelola perangkat teknologi informasi melalui Unit Pengelolaan dan Pelayanan Teknologi Informasi (UP2TI). Masalah ini berdampak pada efisiensi dan efektivitas pengelolaan jaringan. Penelitian ini bertujuan untuk mengevaluasi implementasi *LibreNMS* sebagai sistem manajemen jaringan yang dapat meningkatkan efisiensi pengelolaan oleh tim UP2TI. Metode yang digunakan adalah *Network Development Life Cycle*, dengan analisis terhadap fitur-fitur seperti *alert notification* dan *validate config*. Selain itu, sistem monitoring jaringan berbasis *Simple Network Management Protocol* (SNMP) dengan notifikasi Telegram diimplementasikan untuk mempermudah teknisi dalam pemantauan jaringan. Hasil penelitian menunjukkan bahwa penerapan *LibreNMS* dan sistem notifikasi Telegram meningkatkan efisiensi monitoring dan pengelolaan perangkat jaringan.

**Kata kunci:** *Simple Network Management Protocol; LibreNMS; Network Development Life Cycle; Simple Network Management Protocol*

### 1. Pendahuluan

Pengelolaan jaringan menjadi semakin krusial di tengah kompleksitas lingkungan teknologi informasi saat ini. Perusahaan dan organisasi bergantung pada infrastruktur jaringan mereka untuk menjaga kelancaran operasional, mengamankan data sensitif, dan mendukung produktivitas[1]. Dalam mengelola jaringan, pemantauan terus-menerus terhadap kesehatan dan kinerja jaringan menjadi kunci. Di Fakultas Sains dan Matematika Universitas Diponegoro, Unit Pengelolaan dan Pelayanan Teknologi Informasi (UP2TI) memiliki tanggung jawab utama dalam mengelola dan memberikan layanan terkait teknologi informasi. [2].

Sebagai bagian dari tugasnya, UP2TI harus memantau data *bandwidth* dan ketersediaan perangkat seperti *router*, *switch*, dan *server* di lingkungan fakultas. Proses pemantauan ini, jika dilakukan secara manual, dapat memakan waktu yang cukup lama, dan teknisi hanya dapat mengandalkan laporan dari pengguna untuk mengetahui kondisi jaringan. Situasi ini mengakibatkan keterlambatan dalam mendeteksi kendala pada koneksi internet atau memperbaiki masalah yang terjadi.

*Network Monitoring System* (NMS) dapat menjadi Solusi dalam pengelolaan jaringan komputer, NMS digunakan untuk memantau kualitas bandwidth yang digunakan oleh jaringan dan melibatkan perangkat keras serta perangkat lunak yang mendukung fungsi monitoring [3], [4]. Salah satu protokol yang digunakan adalah *Simple Network Management Protocol* (SNMP) [5], SNMP memiliki fitur seperti pertukaran data, pemantauan jaringan, dan pengelolaan perangkat, terdiri dari tiga elemen utama: manager, agent, dan *management information base* (MIB) [6]. *LibreNMS*, sebuah platform pemantauan jaringan yang menggunakan protokol SNMP, dapat memantau perangkat keras jaringan seperti *switch*, *router*, dan *server*, serta menawarkan fitur-fitur berguna dalam pengelolaan jaringan [7]. Salah satu fitur unggulan *LibreNMS* adalah notifikasi yang dapat terhubung ke *Bot Telegram* [8], memungkinkan teknisi menerima notifikasi secara *real-time* tentang masalah jaringan, sehingga mempercepat penanganan masalah. Implementasi SNMP dan *LibreNMS* di Fakultas Sains dan Matematika Universitas Diponegoro bertujuan untuk memantau traffic bandwidth dan mengetahui ketersediaan perangkat yang terdaftar, meningkatkan efisiensi pengelolaan jaringan, dan memastikan teknisi dapat segera mengetahui dan mengatasi kendala yang terjadi.

Penelitian ini bertujuan untuk mengimplementasikan *LibreNMS* sebagai aplikasi monitoring jaringan di lingkungan Fakultas Sains dan Matematika Universitas Diponegoro. Diharapkan bahwa dengan penggunaan *LibreNMS*, teknisi UP2TI dapat mengelola jaringan dengan lebih efisien dan lebih cepat dalam mendeteksi serta menangani masalah yang mungkin terjadi. Hasil penelitian ini diharapkan dapat meningkatkan efisiensi pekerjaan teknisi dalam pengelolaan jaringan, serta memastikan bahwa jaringan tetap beroperasi secara optimal.

## 2. Tinjauan Pustaka

Penelitian yang dilakukan oleh Danur Wijayanto dan Indra Waspada pada tahun 2016 [2] membahas tentang pengembangan aplikasi *monitoring* jaringan yang menggunakan protokol SNMP dan *Squid Proxy* di UP2TI Fakultas Sains Matematika Universitas Diponegoro. Aplikasi ini membantu admin dalam memantau kondisi perangkat jaringan dan aktivitas pengguna internet. Metode pengembangan yang digunakan adalah *waterfall* dengan PHP dan *framework* CodeIgniter. Implementasi aplikasi ini memungkinkan admin untuk memonitor perangkat jaringan, memantau aktivitas pengguna internet, serta menyimpan traffic masuk dan keluar pada interface perangkat yang dikelola. Referensi yang digunakan dalam penelitian ini antara lain *Essential SNMP 2nd Edition*, *Managing Network Component Using SNMP*, dan *Squid: The Definitive Guide*.

Penelitian yang dilakukan oleh Muhtar Efendy dan Mochamad Mizanul Achlaq pada tahun 2022 [9] membahas mengenai implementasi sistem monitoring dan *backup* konfigurasi perangkat jaringan menggunakan *LibreNMS* di PT. Data Utama Dinamika. Tujuan riset tersebut adalah untuk memantau jaringan dan melakukan pencadangan otomatis. *LibreNMS* digunakan untuk *monitoring* jaringan, sedangkan *Oxidized* digunakan untuk backup konfigurasi perangkat. Hasil penelitian menunjukkan bahwa *LibreNMS* mampu menampilkan grafik *traffic*, informasi perangkat, dan mengirimkan notifikasi melalui Telegram. Selain itu, *backup* konfigurasi perangkat juga berhasil dilakukan dengan *Oxidized*.

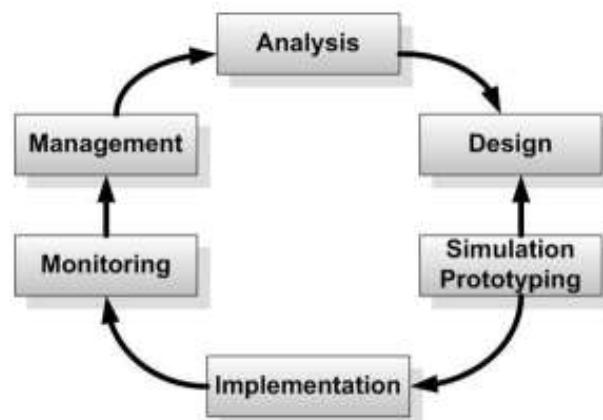
Penelitian yang dilakukan oleh Nendi dan Fath Maulana pada tahun 2024 [10] membahas implementasi sistem pemantauan lalu lintas jaringan berbasis SNMP di jaringan perumahan Permata Puri Harmoni 2. Dengan menggunakan SNMP, penelitian ini bertujuan untuk memantau kinerja jaringan secara *real-time*, meningkatkan efisiensi penggunaan jaringan, dan mengidentifikasi masalah jaringan dengan cepat. Metode penelitian yang digunakan adalah eksperimental dengan pengumpulan data melalui instalasi perangkat lunak *monitoring traffic* pada setiap perangkat yang terhubung. Selain itu, penelitian ini juga membahas pentingnya pemahaman yang akurat terhadap protokol SNMP untuk pengambilan keputusan yang tepat dalam meningkatkan kinerja dan keamanan jaringan. Implementasi *monitoring* jaringan berbasis SNMP berhasil meningkatkan kualitas layanan internet bagi pengguna perumahan tersebut, namun diperlukan pendidikan dan pelatihan tambahan untuk meningkatkan kemampuan administrator jaringan dalam mengelola jaringan dengan efisien.

Penelitian ini menawarkan pendekatan baru dengan menggabungkan implementasi *LibreNMS* untuk monitoring jaringan berbasis SNMP dengan fitur notifikasi melalui *Bot Telegram*. Berbeda dengan riset Wijayanto dan Waspada [2] yang berfokus pada monitoring jaringan menggunakan SNMP dan *Squid Proxy*, serta penelitian Efendy dan Achlaq [9] yang menggabungkan *LibreNMS* dengan *Oxidized* untuk monitoring dan backup konfigurasi, penelitian

ini mengutamakan efisiensi respons teknis melalui notifikasi *real-time*. Selain itu, penelitian ini juga memperluas cakupan penggunaan *LibreNMS* dengan fitur *Auto Discovery* untuk pendaftaran perangkat secara otomatis, memberikan kemudahan dan meningkatkan akurasi dalam manajemen jaringan. Kebaruan riset ini terletak pada integrasi fitur-fitur canggih *LibreNMS* yang belum diterapkan secara komprehensif pada penelitian-penelitian terdahulu, sehingga menawarkan solusi lebih efisien dan responsif dalam pengelolaan jaringan di Fakultas Sains dan Matematika Universitas Diponegoro.

### 3. Metodologi

Penelitian ini menggunakan metode *Network Development Life Cycle* (NDLC) yang terdiri dari beberapa tahap, yaitu *analysis, design, monitoring, implementation, dan management*. NDLC adalah cara mengembangkan sistem jaringan komputer dan memantau kinerjanya, mengandalkan proses pembangunan sebelumnya seperti siklus pengembangan aplikasi, perencanaan strategi bisnis, dan analisis distribusi data [11]. Pada penelitian ini, hanya digunakan lima tahap yaitu *analysis, design, monitoring, implementation, dan management*.



Gambar 1. Siklus NDLC

#### 3.1 Analysis

```

root@librenms:~# neofetch
  .-/+osssoo+/-
  *+ooooooooooooo+*
  +oooooooooooooooo+
  ooooooooooooooooo+
  /oooooooooooooWey/PPPP/
  +oooooooooydNNNNNNHdddyssooooo+
  /oooooooohNNNNhhyyyheNNNN/
  /ooooooooNNNNooooooooNNNN/
  +ssshhhyNNNNooooooooNNNN/
  ossyNNNN/NNHoooooooooooohNNNN/
  ossyNNNN/NNHoooooooooooohNNNN/
  +ssshhhyNNNNooooooooNNNN/
  /oooooooooydNNNNNNHdddyssooooo+
  /oooooooohNNNNhhyyyheNNNN/
  +oooooooooydNNNNNNHdddyssooooo+
  /oooooooohNNNNhhyyyheNNNN/
  ooooooooooooooooo+
  *+ooooooooooooo+*
  .-/+osssoo+/-

root@librenms
-----
OS: Ubuntu 20.04.5 LTS x86_64
Host: 5200CMT
Kernel: 5.15.74-1-pve
Uptime: 18 hours, 11 mins
Packages: 691 (dpkg)
Shell: bash 5.8.17
Resolution: 1920x768
Terminal: /dev/pts/3
CPU: Intel Xeon E5-2640 v4 (6) @ 3.400GHz
GPU: 06:00.0 Matrox Electronics Systems Ltd. MGA G200e [Pilot] ServerEngines
Memory: 366MiB / 4100MiB
  
```

Gambar 2. Sistem operasi yang disediakan

Dalam tahap ini penulis melakukan analisa terhadap sistem dan spesifikasi *Virtual Machine* yang digunakan dalam implementasi sistem monitoring di jaringan Fakultas Sains dan Matematika, berikut adalah hasilnya:

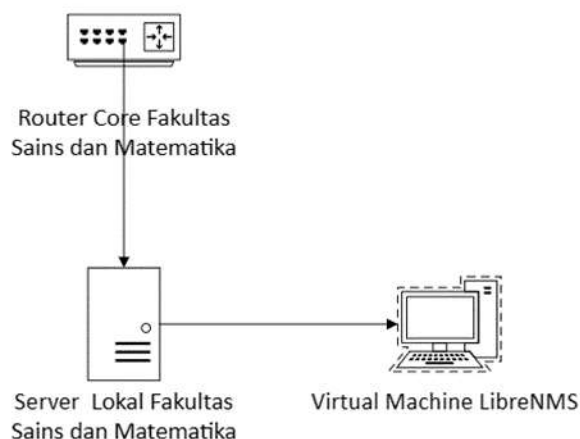
- 1) *Cores*: 6
- 2) *Memory*: 4 GB
- 3) *Storage*: 250 GB

*Virtual Machine* ini disediakan oleh teknisi UP2TI. Pemilihan *Virtual Machine* dibandingkan *server* fisik dilakukan berdasarkan saran dari Kepala Teknisi UP2TI agar sistem hanya dapat diakses secara lokal dari jaringan Fakultas Sains dan Matematika.

Perangkat lunak yang digunakan dalam penelitian ini adalah:

- 1) Sistem Operasi *Server*: Ubuntu 20.04.5 LTS
  - 2) Remote Access *Server*: Terminal SSH
  - 3) Peramban Client: Google Chrome, Microsoft Edge, Mozilla Firefox
- Pemilihan sistem operasi *server* menggunakan Ubuntu 20.04.5 dikarenakan teknisi UP2TI hanya menyediakan *template image* versi tersebut pada saat penelitian dilakukan.

### 3.2 Desain Sistem



Gambar 3. Desain topologi sistem LibreNMS

Pada tahap ini, dilakukan perancangan arsitektur sistem monitoring jaringan berbasis *Simple Network Management Protocol* (SNMP) yang diintegrasikan dengan notifikasi Bot Telegram. Arsitektur sistem ini mencakup komponen-komponen utama sebagai berikut:

- 1) **Perangkat Jaringan (Router, Switch, Server)**: Perangkat-perangkat ini dikonfigurasi untuk mendukung SNMP agar dapat mengirim data kinerja dan status ke sistem monitoring.
- 2) **LibreNMS**: Aplikasi *monitoring* jaringan yang diinstal pada *server* berbasis Ubuntu 20.04.5 LTS. LibreNMS mengumpulkan, menganalisis, dan menampilkan data yang diterima dari perangkat jaringan.
- 3) **Bot Telegram**: Bot yang menerima notifikasi dari LibreNMS mengenai status jaringan dan masalah yang terjadi. Notifikasi ini dikirimkan ke teknisi secara *real-time* untuk memastikan respons cepat terhadap gangguan jaringan.

Desain arsitektur sistem yang diusulkan ini bertujuan untuk meningkatkan efisiensi *monitoring* jaringan dengan memanfaatkan teknologi SNMP dan notifikasi *real-time* melalui Bot Telegram, sehingga mengurangi keterlambatan dalam penanganan masalah jaringan.

### 3.3 Monitoring

*Monitoring* dilakukan untuk memastikan sistem berjalan sesuai dengan desain. Langkah-langkah yang diambil meliputi:

- 1) **Pemantauan performa jaringan**: Menggunakan LibreNMS untuk memantau *traffic bandwidth*, status perangkat, dan performa jaringan.
- 2) **Pengujian notifikasi**: Mengatur dan menguji notifikasi Bot Telegram untuk memastikan teknisi menerima informasi secara *real-time*.

### 3.4 Implementation

Implementasi mencakup instalasi dan konfigurasi sistem:

- 1) **Instalasi LibreNMS**: Mengikuti panduan instalasi untuk menginstal LibreNMS pada Ubuntu 20.04.5 LTS.
- 2) **Konfigurasi SNMP**: Mengatur perangkat jaringan untuk menggunakan protokol SNMP agar dapat diintegrasikan dengan LibreNMS.

- 3) **Integrasi Bot Telegram:** Mengkonfigurasi *LibreNMS* untuk mengirim notifikasi ke Bot Telegram [12].

### 3.5 Management

Tahap terakhir adalah manajemen sistem untuk memastikan sistem beroperasi dengan optimal:

- 1) **Pemeliharaan sistem:** Melakukan *update* dan *maintenance* rutin pada *LibreNMS* dan server untuk memastikan kinerja yang optimal.
- 2) **Pelatihan teknisi:** Memberikan pelatihan kepada teknisi UP2TI tentang cara menggunakan sistem *monitoring* dan menangani notifikasi dari *Bot Telegram*.
- 3) **Evaluasi dan peningkatan:** Secara berkala mengevaluasi kinerja sistem dan mengidentifikasi area untuk peningkatan.

## 4. Hasil dan Pembahasan

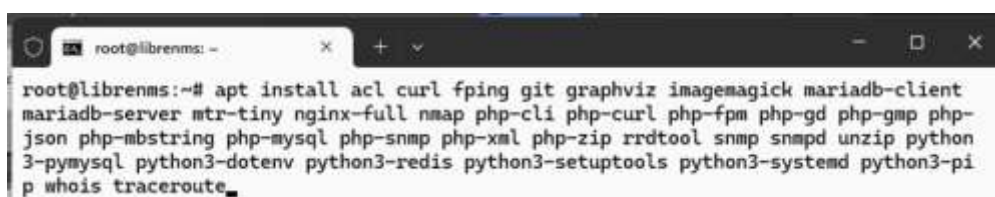
### 4.1. Implementasi

Implementasi diawali dengan melakukan uji coba sistem operasi yang disediakan oleh teknisi UP2TI. Sistem operasi yang di sediakan adalah Ubuntu 20.04.5 LTS x86\_64. Spesifikasi dari *Virtual Machine* yang disediakan memiliki 6 core, 4 GB ram, dan 250 GB Storage

#### 4.1.1. Instalasi LibreNMS

Berikut adalah langkah-langkah yang dilakukan penulis dalam instalasi *LibreNMS*, yang mengacu pada langkah instalasi yang disediakan oleh *LibreNMS* [13]:

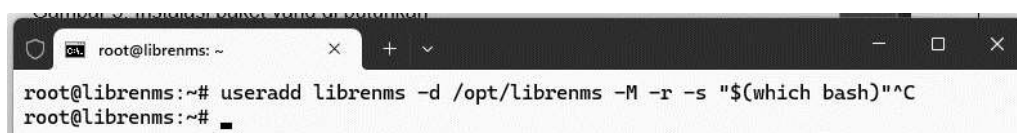
- 1) Penulis melakukan instalasi paket yang dibutuhkan



```
root@librenms:~# apt install acl curl fping git graphviz imagemagick mariadb-client mariadb-server mtr-tiny nginx-full nmap php-cli php-curl php-fpm php-gd php-gmp php-json php-mbstring php-mysql php-snmp php-xml php-zip rrdtool snmp snmpd unzip python3-pymysql python3-dotenv python3-redis python3-setuptools python3-systemd python3-pip whois traceroute
```

Gambar 4. Instalasi paket yang di butuhkan

- 2) Selanjutnya dilakukan penambahan user librenms pada sistem operasi



```
root@librenms:~# useradd librenms -d /opt/librenms -M -r -s "$(which bash)"^C
root@librenms:~#
```

Gambar 5. Penambahan user librenms

- 3) Selanjutnya penulis mengunduh LibreNMS
 

```
cd /opt
git clone https://github.com/librenms/librenms.git
```
- 4) Selanjutnya penulis melakukan konfigurasi *permissions* pada sistem operasi



```
root@librenms:~# chown -R librenms:librenms /opt/librenms
root@librenms:~# chmod 771 /opt/librenms
root@librenms:~# setfacl -d -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/ /opt/librenms/storage/
root@librenms:~# setfacl -R -m g::rwx /opt/librenms/rrd /opt/librenms/logs /opt/librenms/bootstrap/cache/ /opt/librenms/storage/
root@librenms:~#
```

Gambar 6. Konfigurasi *permissions*

- 5) Selanjutnya dilakukan instalasi PHP *dependencies*

```
su - librenms
./scripts/composer_wrapper.php install --no-dev
Exit
```

- 6) Selanjutnya penulis melakukan konfigurasi zona waktu  
 nano /etc/php/8.1/fpm/php.ini

```

; Module Settings ;

[CLI Server]
; Whether the CLI web server uses ANSI color coding in its terminal output.
cli_server.color = On

[Date]
; Defines the default timezone used by the date functions
; https://php.net/date.timezone
date.timezone = "Asia/Jakarta"
    
```

Gambar 7. Konfigurasi pada zona waktu di /etc/php/8.1/fpm/php.ini

nano /etc/php/8.1/cli/php.ini

```

; Module Settings ;

[CLI Server]
; Whether the CLI web server uses ANSI color coding in its terminal output.
cli_server.color = On

[Date]
; Defines the default timezone used by the date functions
; https://php.net/date.timezone
date.timezone = "Asia/Jakarta"
    
```

Gambar 8. Konfigurasi pada zona waktu di /etc/php/8.1/cli/php.ini

Setelah itu lakukan konfigurasi pada sistem operasi.

```

root@librenms:~# timedatectl set-timezone Asia/Jakarta
    
```

Gambar 9. Konfigurasi zona waktu di sistem operasi

- 7) Konfigurasi penulis pada MariaDB.  
 nano /etc/mysql/mariadb.conf.d/50-server.cnf

Di tahap ini penulis menambahkan di bawah bagian [mysqld].

```

[mysqld]
innodb_file_per_table=1
lower_case_table_names=0
    
```

Gambar 10. Konfigurasi pada bagian [mysqld]

Setelah penulis melakukan konfigurasi MariaDB, penulis melakukan *restart service* pada *service* MariaDB, penulis melakukan konfigurasi pada MariaDB.

mysql -u root

Tulis perintah untuk membuat database, user, dan konfigurasi privileges ke user yang telah dibuat sebelumnya.

```

MariaDB [(none)]> CREATE DATABASE librenms CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE USER 'librenms'@'localhost' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.004 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON librenms.* TO 'librenms'@'localhost';
Query OK, 0 rows affected (0.013 sec)
    
```

Gambar 11. Konfigurasi pada MariaDB

- 8) Selanjutnya penulis melakukan konfigurasi pada PHP-FPM.  
 cp /etc/php/8.1/fpm/pool.d/www.conf /etc/php/8.1/fpm/pool.d/librenms.conf  
 nano /etc/php/8.1/fpm/pool.d/librenms.conf

Di tahap ini penulis melakukan perubahan pada [www] menjadi [librenms], user dan group menjadi librenms, dan listen menjadi listen = /run/php-fpm-librenms.sock.

```
[librenms]

; Per pool prefix
; It only applies on the following directives:
; - 'access.log'
; - 'slowlog'
; - 'listen' (unixsocket)
; - 'chroot'
; - 'chdir'
; - 'php_values'
; - 'php_admin_values'
; When not set, the global prefix (or /usr) applies instead.
; Note: This directive can also be relative to the global prefix.
; Default Value: none
;prefix = /path/to/pools/$pool

; Unix user/group of processes
; Note: The user is mandatory. If the group is not set, the default user's group
; will be used.
user = librenms
group = librenms

; The address on which to accept FastCGI requests.
; Valid syntaxes are:
; 'ip.add.re.ss:port'   - to listen on a TCP socket to a specific IPv4 address on
;                        a specific port;
; '[ip6:addr:ess]:port' - to listen on a TCP socket to a specific IPv6 address on
;                        a specific port;
; 'port'                - to listen on a TCP socket to all addresses
;                        (IPv6 and IPv4-mapped) on a specific port;
; '/path/to/unix/socket' - to listen on a unix socket.
; Note: This value is mandatory.
listen = /run/php-fpm-librenms.sock
```

Gambar 12. Konfigurasi pada PHP-FPM

- 9) Konfigurasi penulis pada Web Server.  
nano /etc/apache2/sites-available/librenms.conf

Penulis menambahkan *script* dibawah ini kedalam file tersebut.



```
GNU nano 4.8 /etc/apache2/sites-available/librenms.conf
VirtualHost *:80>
DocumentRoot /opt/librenms/html/
ServerName librenms.fsm.undip.ac.id

AllowEncodedSlashes NoDecode
<Directory "/opt/librenms/html/">
  Require all granted
  AllowOverride All
  Options FollowSymLinks MultiViews
</Directory>

# Enable http authorization headers
<IfModule setenvif_module>
  SetEnvIfNoCase "Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
</IfModule>

<FilesMatch "\.php$">
  SetHandler "proxy:unix:/run/php-fpm-librenms.sock|fcgi://localhost"
</FilesMatch>
</VirtualHost>
```

Gambar 13. Konfigurasi apache2 librenms.conf

Langkah selanjutnya dilakukan adalah menonaktifkan *default apache*, mengaktifkan *mod apache* yang diperlukan, mengaktifkan konfigurasi apache LibreNMS dan melakukan restart pada apache2 dan PHP

```

root@librenms:~# a2dissite 000-default
Site 000-default already disabled
root@librenms:~# a2enmod proxy_fcgi setenvif rewrite
Considering dependency proxy for proxy_fcgi:
Module proxy already enabled
Module proxy_fcgi already enabled
Module setenvif already enabled
Module rewrite already enabled
root@librenms:~# a2ensite librenms.conf
Site librenms already enabled
root@librenms:~# systemctl restart apache2
root@librenms:~# systemctl restart php8.1-fpm
root@librenms:~#

```

Gambar 14. Konfigurasi apache2

- 10) Selanjutnya penulis menambahkan *file lnms command completion*. Fitur ini memberikan kesempatan untuk menggunakan tab untuk perintah lnms dan sama dengan perintah linux.
- ```
ln -s /opt/librenms/lnms /usr/bin/lnms
cp /opt/librenms/misc/lnms-completion.bash /etc/bash_completion.d/
```
- 11) Setelah itu penulis melakukan konfigurasi pada snmpd.
- ```
cp /opt/librenms/snmpd.conf.example /etc/snmp/snmpd.conf
nano /etc/snmp/snmpd.conf
```

Penulis merubah bagian *RANDOMSTRINGGOESHERE* menjadi *up2ti-community*.

```

GNU nano 6.2
$ Change RANDOMSTRINGGOESHERE to your preferred SNMP community string
com2sec readonly default up2ti-community

group MyROGroup v2c      readonly
view all included .1    80
access MyROGroup ""     any      noauth  exact  all   none  none

syslocation Rack, Room, Building, City, Country [Lat, Lon]
syscontact Your Name <your@email.address>

#OS Distribution Detection
extend distro /usr/bin/distro

#Hardware Detection
$ (uncomment for x86 platforms)
#extend manufacturer '/bin/cat /sys/devices/virtual/dmi/id/sys_vendor'
#extend hardware '/bin/cat /sys/devices/virtual/dmi/id/product_name'
#extend serial '/bin/cat /sys/devices/virtual/dmi/id/product_serial'

$ (uncomment for ARM platforms)
#extend hardware '/bin/cat /sys/firmware/devicetree/base/model'
#extend serial '/bin/cat /sys/firmware/devicetree/base/serial-number'

```

Gambar 15. Konfigurasi pada snmpd.conf

Setelah menyimpan konfigurasi sebelumnya, di tahap ini penulis menjalankan perintah berikut.

```
curl -o /usr/bin/distro https://raw.githubusercontent.com/librenms/librenms-agent/master/snmp/distro
chmod +x /usr/bin/distro
```

Setelah itu dilakukan *restart* pada *service snmpd*.

- 12) Selanjutnya penulis menambahkan *Cron Job librenms* dan *scheduler* pada sistem operasi.



```

root@librenms:~# cp /opt/librenms/dist/librenms.cron /etc/cron.d/librenms
root@librenms:~# cp /opt/librenms/dist/librenms-scheduler.service /opt/librenms/dist/librenms-scheduler.timer /etc/systemd/system/
root@librenms:~# systemctl enable librenms-scheduler.timer
Created symlink /etc/systemd/system/timers.target.wants/librenms-scheduler.timer -> /etc/systemd/system/librenms-scheduler.timer.
root@librenms:~# systemctl start librenms-scheduler.timer
root@librenms:~#

```

Gambar 16. Konfigurasi Cron Job dan *scheduler*

- 13) Selanjutnya penulis menyalin *file* konfigurasi logrotate.  
`cp /opt/librenms/misc/librenms.logrotate /etc/logrotate.d/librenms`

#### 4.1.2. Konfigurasi pada Web LibreNMS

Langkah berikutnya adalah konfigurasi penulis pada LibreNMS pada web *browser*. Berikut langkah-langkah yang dilakukan penulis untuk melakukan konfigurasinya:

- 1) Langkah pertama penulis membuka *browser* lalu memasukkan *Ip Address* atau *domain* yang telah di konfigurasi di sistem operasi. Penulis di berikan *Ip Address* oleh teknisi UP2TI 10.137.58.100. Setelah itu alamat tersebut ditambahkan */install* menjadi 10.137.58.100/install.



Gambar 17. Tampilan pertama */install* pada LibreNMS

- 2) Selanjutnya penulis melakukan konfigurasi *database* yang telah di buat pada tahapan sebelumnya.



Gambar 18. Konfigurasi pada Database halaman web

- 3) Setelah itu penulis membuat *Username* dan *Password* untuk *Administrator*.



Gambar 19. Pembuatan *Username* dan *Password* pada halaman web

- 4) Setelah berhasil melakukan instalasi, tampilan *LibreNMS* berubah menjadi tampilan halaman Login.



Gambar 20. Tampilan halaman login pada *LibreNMS*



Gambar 21. Tampilan awal *LibreNMS*

#### 4.1.3. Cara Menambahkan Perangkat Kedalam LibreNMS

Langkah penulis berikutnya adalah untuk menambahkan perangkat yang akan di monitoring ke dalam LibreNMS [14]. Penulis diberikan 1 perangkat sebagai perangkat yang akan di daftarkan ke dalam LibreNMS oleh teknisi dari UP2TI.

##### **ARUBA 6100 48G CL4 4SFP+ Switch JL675A**

- 1) *Switch managed*
- 2) *48 ports Gigabit*
- 3) *Switching Capacity 176 Gbps*
- 4) *Fixed power supply (500W), Supports PoE Standards IEEE 802.3af; 802.3at*



Gambar 22. Switch Aruba 48G CL4 4SFP+ JL675A

Berikut ini tahapan yang penulis lakukan untuk mendaftarkan perangkat tersebut ke dalam aplikasi LibreNMS:

- 1) Penulis membuka *Web Client Switch* pada *browser*, lalu membuka menu *System* lalu *SNMP*



Gambar 23. Menu Switch Aruba 48G CL4 4SFP+ JL675A

- 2) Selanjutnya penulis menambahkan *SNMP Community* yang telah dibuat.



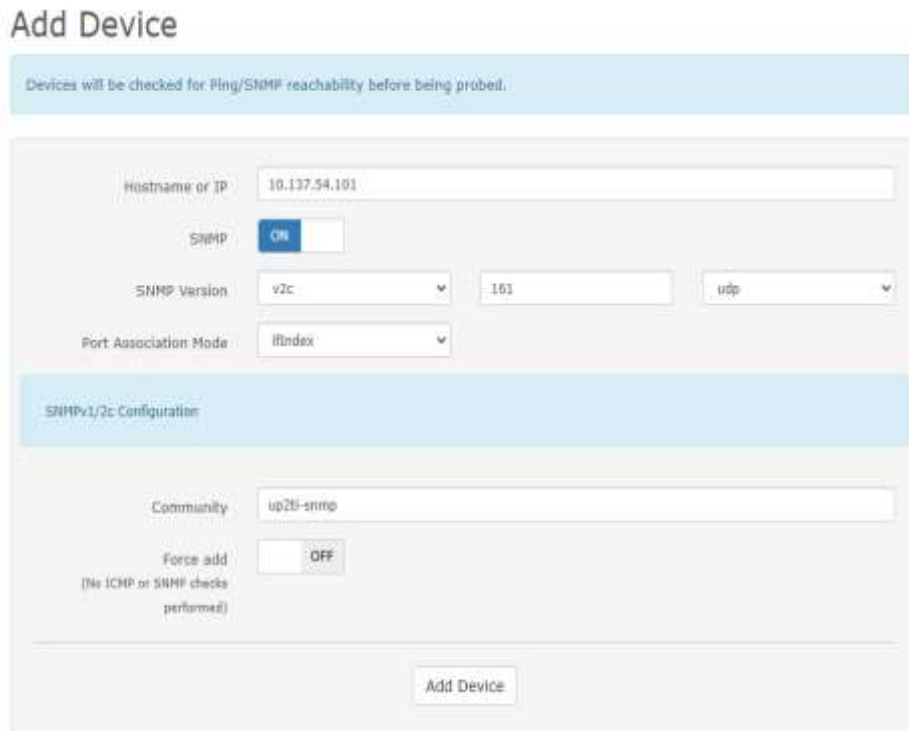
Gambar 24. Menambahkan *SNMP Community* pada perangkat.

- 3) Selanjutnya penulis melakukan konfigurasi pada *Trap Receivers* LibreNMS.



Gambar 25. Menambahkan *Trap Host* pada perangkat.

- 4) Setelah itu penulis melanjutkan menambahkan perangkat yang sudah di konfigurasi pada tahap sebelumnya kedalam *LibreNMS*.

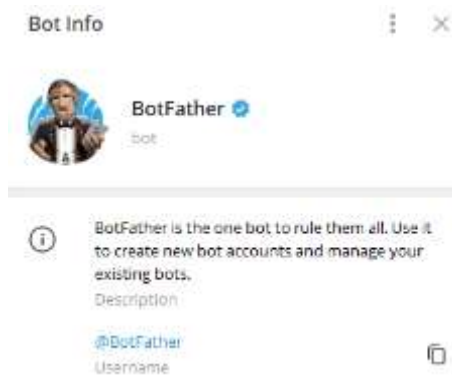


Gambar 26. Menambahkan perangkat kedalam LibreNMS.

#### 4.1.4. Konfigurasi penulis pada fitur Notification Alert

Langkah selanjutnya penulis melakukan konfigurasi pada fitur *Alert* pada LibreNMS guna memberikan notifikasi kepada teknisi UP2TI melalui aplikasi Telegram. Berikut adalah langkah untuk konfigurasi pada Telegram dan *LibreNMS*:

- 1) Untuk mendapatkan notifikasi pada aplikasi telegram, *LibreNMS* harus terhubung dengan *Chat Bot* pada telegram [15]. Penulis menggunakan BotFather untuk membuat *Chat Bot*.



Gambar 27. Akun BotFather.

- 2) Selanjutnya penulis mengikuti instruksi dari BotFather untuk pembuatan *Chat Bot*, disini penulis membuat *Chat Bot* baru dengan *username* librenmsup2ti\_bot dan memberikan foto profil *Chat Bot* tersebut.



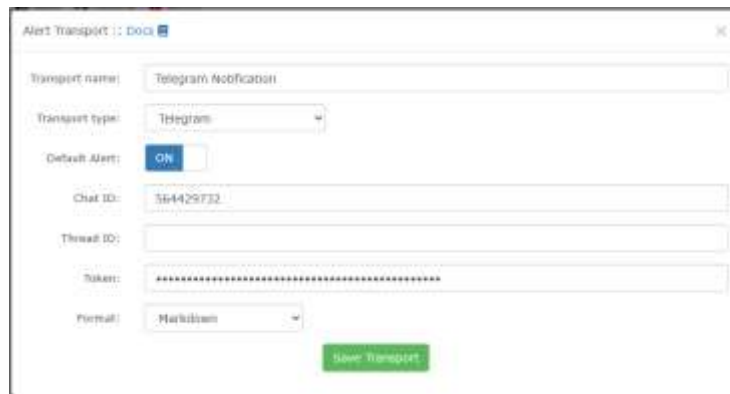
Gambar 28. Pembuatan *Chat Bot* pada BotFather.

- 3) Selanjutnya penulis melakukan konfigurasi *Alert Transports* pada LibreNMS. Data yang dibutuhkan adalah *Chat ID* dan Token yang sudah di *generate* oleh BotFather. Untuk mendapatkan *Chat ID* penulis membuka <https://api.telegram.org/bot6439082928:AAFClp75VWlxuTo829g0BJEbw-iKtTuvW78/getUpdates> dan menulis pesan pada *Chat Bot* yang sudah dibuat tadi. Pada alamat *website* tersebut akan tertera *Chat ID* yang penulis butuhkan



Gambar 29. Data *Chat ID Chat Bot*.

- 4) Setelah itu penulis membuat *alert transport* lalu menuliskan data yang dibutuhkan.



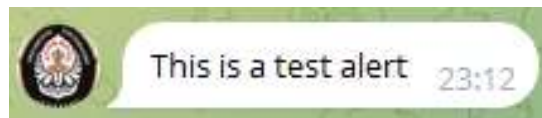
Gambar 30. Menambahkan *Alert Transport* Telegram kedalam LibreNMS.

- 5) Setelah itu penulis melakukan uji coba pada fitur tersebut dengan menekan tombol *Test Transport*.

Transport Name	Transport Type	Default	Details	Action
Telegram Notification	Telegram	Yes	Chat ID: 564429732 Thread ID: Token: ***** Format: Markdown	

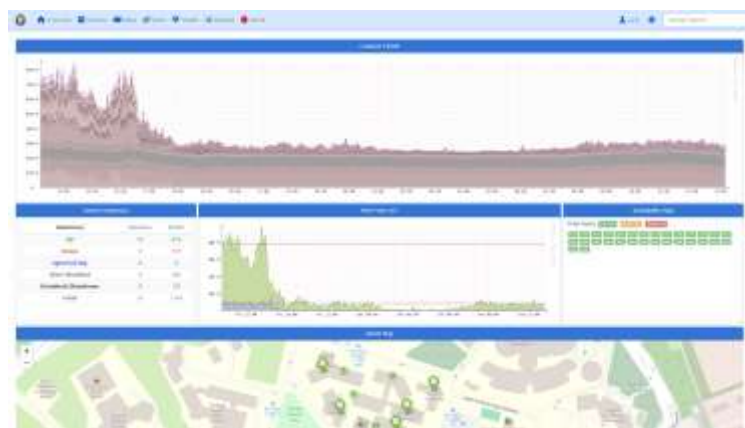
Gambar 31. *Alert transport* notifikasi telegram.

- 6) Selanjutnya penulis membuka kembali *Chat Bot* yang sudah dibuat tadi, jika *test alert* berhasil terkirim, artinya koneksi dari *LibreNMS* ke *Telegram Chat Bot* berhasil.



Gambar 32. Pesan alert transport notifikasi telegram.

#### 4.2. Pembahasan



Gambar 4. Tampilan LibreNMS.

Implementasi sistem monitoring jaringan berbasis SNMP dengan notifikasi Bot Telegram telah berhasil dilakukan. Tahapan-tahapan proses monitoring yang dilakukan mencakup pengumpulan data kinerja dan status perangkat jaringan menggunakan SNMP, analisis data, serta pengiriman notifikasi *real-time* melalui Bot Telegram kepada teknisi UP2TI.

Pembahasan terfokus pada sejauh mana konsep yang diusulkan dapat menyelesaikan masalah yang telah diidentifikasi sebelumnya, yaitu keterlambatan dalam penanganan masalah

jaringan. Hasil temuan dalam penelitian ini memberikan kontribusi dalam mempercepat respons teknisi terhadap gangguan jaringan dengan memberikan notifikasi secara *real-time*.

Selain itu, relevansi antara hasil temuan dalam penelitian ini dengan hasil-hasil temuan pada penelitian terdahulu juga dibahas. Temuan ini memperkuat temuan pada penelitian terdahulu yang mendukung penggunaan SNMP sebagai protokol monitoring jaringan. Referensi-referensi yang mendukung temuan ini telah disertakan dalam tinjauan pustaka.

Device Summary		
Summary	Devices	Ports
Up	30	630
Down	0	738
Ignored tag	0	0
Alert disabled	0	NA
Disabled/Shutdown	0	15
<b>Total</b>	<b>30</b>	<b>1383</b>

Gambar 5. Tampilan Device Summary

Tampilan Device Summary pada *LibreNMS* memberikan informasi tentang status perangkat jaringan yang terhubung, termasuk perangkat yang sedang aktif dan tidak aktif.



Gambar 6. Tampilan LibreNMS tentang *bandwidth* dari perangkat

Grafik monitoring *bandwidth* menampilkan penggunaan *bandwidth* dari perangkat jaringan selama periode waktu tertentu.



Gambar 7. Notifikasi perangkat pada *Chat Bot Telegram*.

Notifikasi masalah jaringan dikirimkan secara *real-time* melalui *Bot Telegram* kepada teknisi UP2TI, memungkinkan respons cepat terhadap gangguan yang terjadi.

## 5. Simpulan

Setelah menganalisis implementasi LibreNMS pada Fakultas Sains dan Matematika, ditemukan bahwa aplikasi ini dapat secara signifikan mempermudah tim UP2TI dalam mengelola jaringan. Dibandingkan dengan aplikasi sebelumnya yang dikembangkan pada tahun 2016, *LibreNMS* menonjol dengan fitur-fitur canggih seperti *Validate Config* yang mendukung

pengelolaan sistem. Dari wawancara dengan Kepala UP2TI, diketahui bahwa *LibreNMS* memiliki antarmuka yang mudah dipahami, sementara statusnya sebagai aplikasi *Open Source* membantu mengurangi biaya pengembangan. Hal ini menegaskan bahwa *LibreNMS* bukan hanya sebuah solusi efektif sebagai *Network Management System* (NMS), tetapi juga dapat diterapkan secara sukses dalam konteks jaringan Fakultas Sains dan Matematika. Namun, pentingnya menyertakan referensi dalam simpulan untuk memperkuat klaim dan menegaskan relevansi temuan dengan penyelesaian masalah perlu ditekankan. Selain itu, simpulan harus menggarisbawahi temuan utama penelitian ini dan memberikan wawasan tentang prospek pengembangan masa depan dari hasil yang diperoleh.

#### Daftar Referensi

- [1] H. Kuswanto dan T. Rahman, "Failover Gateway Menggunakan Protokol Virtual Router Redundancy Protocol (VRRP) pada Mikrotik Router," *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 7, no. 1, pp. 1-12, 2019, doi: 10.26418/justin.v7i1.29482.
- [2] D. Wijayanto dan I. Waspada, "Aplikasi Monitoring Perangkat dan Aktivitas Pengguna pada Jaringan Menggunakan Protocol SNMP dan Squid proxy," *Jurnal Teknologi dan Sistem Informasi*, vol. 2, pp. 11-20, Des 2016, doi: 10.25077/TEKNOSI.v2i3.2016.11-20.
- [3] H. Annur dan R. A. Laari, "Penerapan Network Monitoring System (Nms) Secara Visual Pada Infrastruktur Jaringan Fisik Berbasis WEB," *Nusantara of Engineering (NOE)*, vol. 5, no. 2, pp. 45-58, 2022, doi: 10.29407/noe.v5i2.18682.
- [4] R. Rino dan E. M. Teja, "Network Tools Application Design Based on C# with Intrusion Detection Methods and Simple Network Management Protocol," *bit-Tech*, vol. 3, no. 3, pp. 22-34, 2021, doi: 10.32877/bt.v3i3.183.
- [5] P. Purnomo, M. Nugroho, M. S. Kabes, S. P. Putra, dan J. Fathanah, "Sistem Pemantauan Jaringan Data Di Stasiun Bumi LAPAN," *Format Jurnal Ilmiah Teknik Informatika*, vol. 11, no. 1, pp. 5-15, 2022, doi: 10.22441/10.22441/format.2022.v11.i1.004.
- [6] I. Vingestin, T. U. Kalsum, dan Y. Mardiana, "The Design Of Network Monitoring System Using SNMP Protocol With Telegram Notification," *Jurnal Media Computer Science*, vol. 2, no. 1, pp. 17-29, 2023, doi: 10.37676/jmcs.v2i1.3441.
- [7] W. Panji Christyan Wijaya dan M. Mizanul Achlaq, "Implementasi Api Bot Telegram Untuk Sistem Notifikasi LibreNms Pada Perusahaan Blip Integrator," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, pp. 50-61, 2024, doi: 10.36040/jati.v7i6.7991.
- [8] "Features - LibreNMS Docs," LibreNMS, 17 April 2024. [Online]. Tersedia pada: <https://docs.librenms.org/Support/Features/>. [Diakses: 19 Juli 2024].
- [9] M. Efendy dan M. Achlaq Mizanul, "Implementasi Sistem Monitoring Dan Backup Konfigurasi Perangkat Jaringan Menggunakan LibreNms Di Pt. Data Utama Dinamika," *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 11, no. 4, pp. 55-66, 2022, doi: 10.30591/smartcomp.v11i4.4253.
- [10] N. Nendi dan F. Maulana, "Monitoring Traffic Berbasis SNMP pada Jaringan Perumahan Permata Puri Harmoni 2," *Jurnal Sains dan Teknologi*, vol. 5, no. 3, pp. 25-36, 2024, doi: 10.55338/saintek.v5i3.1346.
- [11] N. R. Fachrurrozi, A. A. Wirabudi, dan S. A. Rozano, "Design of network monitoring system based on LibreNMS using Line Notify, Telegram, and Email notification," *Sinergi (Indonesia)*, vol. 27, no. 1, pp. 14-27, 2023, doi: 10.22441/sinergi.2023.1.013.
- [12] T. Purnama, Y. Muhyidin, dan D. Singasatia, "Implementasi Intrusion Detection System (Ids) Snort Sebagai Sistem Keamanan Menggunakan Whatsapp Dan Telegram Sebagai Media Notifikasi," *Jurnal Teknologi Informasi Dan Komunikasi*, vol. 14, no. 2, pp. 35-47, 2023, doi: 10.51903/jtikp.v14i2.726.
- [13] "Installing LibreNMS - LibreNMS Docs," LibreNMS, 8 Juni 2024. [Online]. Tersedia pada: <https://docs.librenms.org/Installation/Install-LibreNMS/>. [Diakses: 19 Juli 2024].
- [14] I. W. Krisna Saputra, D. M. Wiharta, dan N. Putra Sastra, "Implementasi Sistem Pemantauan Jaringan Menggunakan LibreNms Pada Jaringan Kampus Universitas Udayana," *Jurnal SPEKTRUM*, vol. 7, no. 2, pp. 75-89, 2020, doi: 10.24843/spektrum.2020.v07.i02.p11.
- [15] A. D. Mulyanto, "Pemanfaatan Bot Telegram Untuk Media Informasi Penelitian," *MATICS*, vol. 12, no. 1, pp. 22-33, 2020, doi: 10.18860/mat.v12i1.8847.