

IMPLEMENTASI API BOT TELEGRAM UNTUK SISTEM NOTIFIKASI LIBRENMS PADA PERUSAHAAN BLIP INTEGRATOR

Wahyu Panji Christyan Wijaya, Mochamad Mizanul Achlaq

Sistem Informasi, Universitas Narotama Surabaya

Jalan Arief Rachman Hakim 51, Sukolilo Surabaya, Indonesia

Wahyupanji1725@gmail.com

ABSTRAK

Teknologi Informasi lebih mudah dan cepat untuk didapat kapan saja dan dimana saja, Teknologi pada dasarnya memiliki tujuan untuk mempermudah manusia dalam melakukan sesuatu. Integrasi merupakan proses dalam komunikasi beberapa aplikasi seperti media sosial dengan aplikasi chatting lainnya salah satunya aplikasi telegram yang dapat difungsikan sebagai aplikasi chatting serta salah satu fiturnya yaitu Bot Telegram. Masalah jaringan yang sering di alami Perusahaan BLiP Integrator adalah keterbatasannya tenaga operasional pada tim yang mengakibatkan masalah pada operasional jaringan khususnya pada instansi yang memanfaatkan adanya jaringan tersebut. Penelitian ini bertujuan untuk mendeskripsikan penggunaan Telegram Bot yang diintegrasikan dengan LibreNMS untuk membantu administrator dalam melakukan aktivitas pemantauan jaringan serta penyelesaian gangguan. Metode yang digunakan yaitu deskriptif dengan pendekatan kualitatif. Subjek penelitian ini merupakan administrator (tim helpdesk dan tim technical support) pada Perusahaan BLiP. Berdasarkan hasil dari penelitian ini dapat disimpulkan bahwa dengan mengintegrasikan Bot Telegram dan LibreNMS dapat membantu administrator dalam memantau jaringan dengan menampilkan status trouble dan recovery berupa alert notification Up dan Down. Dengan alert notification yang mencakup Bot pada telegram, alert rule, dan LibreNMS maka kegiatan pemantauan jaringan dapat dilakukan dengan lebih fleksibel dimanapun dan kapanpun sehingga penanganan gangguan jaringan menjadi lebih cepat dan tepat dalam mengurangi waktu gangguan.

Kata kunci: Telegram, Bot, Notification Alert, LibreNMS, Pemantauan Jaringan, Perusahaan BLiP Integrator

1. PENDAHULUAN

Teknologi Informasi lebih mudah dan cepat untuk didapat kapan saja, dimana saja, Teknologi pada dasarnya memiliki tujuan yaitu untuk mempermudah manusia dalam melakukan sesuatu[1]. Integrasi merupakan proses dalam komunikasi beberapa aplikasi seperti media sosial dengan aplikasi chatting lainnya salah satunya aplikasi telegram yang dapat difungsikan sebagai aplikasi chatting serta salah satu fiturnya yaitu Bot Telegram. Bot adalah aplikasi third party yang berjalan pada aplikasi Telegram sehingga dapat dimanfaatkan pada berbagai aplikasi seperti asisten messenger serta dapat difungsikan sebagai system notification alert pada perusahaan di bidang Teknologi informasi dan Internet Service Provider. Bot dapat dimanfaatkan sebagai program otomatis untuk memudahkan pekerjaan.

Perusahaan BLiP Integrator merupakan perusahaan swasta sebagai penyedia layanan jaringan interkoneksi yang bekerja secara profesional, cepat dan handal pada penanganan jaringan yang dikelolanya, ketika jaringan putus ke customer pada wilayah tertentu maka customer menghubungi nomor customer service bagian administrator jaringan yang pada realitanya para administrator jaringan tidak dapat berada dikantor setelah diluar jam kerja dalam memantau jaringan, dengan terbatasnya tenaga operasional pada tim yang ada di Perusahaan BLiP Integrator mengakibatkan masalah pada operasional jaringan yang tidak kecil khususnya pada instansi-instansi yang memanfaatkan adanya jaringan tersebut, dengan adanya keterbatasan tersebut maka

pemantauan jaringan membutuhkan sistem Network Monitoring System (NMS) untuk dapat mempermudah proses menemukan permasalahan yang ada sehingga dapat lebih cepat dalam melakukan perbaikan sistem[2], namun sistem monitoring pada LibreNMS yang digunakan masih bersifat manual monitoring, tim kesulitan dalam melakukan penelusuran kendala pada device yang dimonitoring sehingga penanganan lambat, maka diperlukan suatu aplikasi yang dapat memberikan informasi secara otomatis berupa system notification alert untuk mengetahui status perangkat sedang up atau down.

Berdasarkan pembahasan latar belakang serta permasalahan yang ada, maka dalam penelitian yang berjudul “Implementasi API Bot Telegram Untuk Sistem Notifikasi LibreNMS Pada Perusahaan Blip Integrator” ini akan membahas system notification alert dari telegram yang terintegrasi dengan LibreNMS sehingga dapat mempermudah proses pemantauan dan perbaikan lebih fleksibel, tepat dan cepat tanpa harus berada di lingkungan kantor[3], serta menerapkan notifikasi informasi status perangkat dan gangguan jaringan yang dapat diterima secara real time.

2. TINJAUAN PUSTAKA

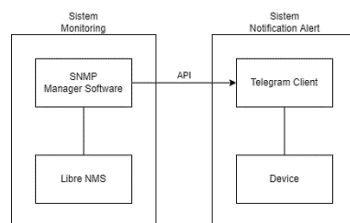
Monitoring jaringan merupakan proses memantau jaringan secara terus-menerus untuk mengetahui informasi besar kecilnya traffic yang sedang berjalan pada suatu jaringan, masalah jaringan internal seperti router yang mengalami full traffic, kegagalan server. Dalam sistem pemantauan ini nantinya akan ada fitur grafik, report (laporan), dengan

banyak fitur tersebut untuk memudahkan kita memantau penggunaan bandwidth[4], Tujuan dari monitoring sistem jaringan adalah untuk mengoptimalkan serta memastikan bahwa perangkat jaringan bekerja dengan baik dan dapat mencegah gangguan yang ada, serta meminimalisir waktu downtime ketika terjadi gangguan, Network Management System (NMS) bertanggung jawab ketika polling dijalankan dan menerima trap dari perangkat jaringan yang terhubung. Simple Network Management Protocol (SNMP) terdiri dari dua bagian yaitu SNMP Manager atau Network Management Station dan SNMP agent yang banyak digunakan sebagai monitoring kinerja jaringan[5],

Telegram dapat difungsikan sebagai *system notification alert* menggunakan integrasi API, salah satu API yang disediakan adalah fitur Bot[6], Bot dapat dimanfaatkan sebagai program otomatis untuk memudahkan perusahaan dalam memantau status perangkat jaringan.

2.1. Transport Layer

Transport Layer adalah salah satu dari OSI (Open Systems Interconnection) layer dan juga model TCP/IP (Transmission Control Protocol/Internet Protocol) bertugas mengatur koneksi antara dua host yang berjalan pada jaringan yang berbeda, pada proses transport yang terjadi menggunakan sesi koneksi yang bersifat connection-oriented atau broadcast yang bersifat connectionless, TCP/Transmission Control Protocol dan UDP/User Datagram Protocol merupakan proses dari layer ini.



Gambar 1 Struktur proses transport data

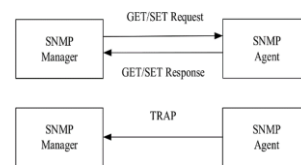
Transport data pada Gambar 1 dijelaskan tahap dalam pengiriman data dari data sistem monitoring libreNMS ke sistem peringatan Bot telegram dengan menggunakan protokol atau metode tertentu, dalam proses Transport Data yang digunakan ada dua sistem utama yang akan diterapkan, yaitu Sistem Monitoring dan Sistem Notification Alert, API token yang diberikan Bot pada Telegram dapat digunakan untuk mengintegrasikan aplikasi LibreNMS alert transport nantinya akan dilakukan pengujian mengirimkan peringatan[7] dari LibreNMS ke grup Telegram, untuk memberikan perintah pada BOT Telegram diperlukan konfigurasi pengiriman notifikasi ke BOT tersebut sehingga pengguna dapat melakukan monitoring secara real-time dari Telegram.

2.2. Sistem Alerting

Sistem alerting pada sistem monitoring jaringan menjadi suatu mekanisme untuk memberikan notifikasi atau pemberitahuan kepada pengguna atau administrator ketika terjadi masalah atau gangguan dalam jaringan. Telegram Bot API memungkinkan siapa saja untuk membuat bot yang akan membalas semua penggunaannya jika mengirimkan pesan perintah yang dapat diterima oleh Bot tersebut[6].

2.3. SNMP

SMTP (Simple Mail Transfer Protocol) adalah protokol aplikasi yang berfungsi untuk proses transfer surat elektronik. Standard protokol ini didefinisikan dalam RFC 821 dan disempurnakan pada RFC 5321[8].



Gambar 2 Proses SNMP

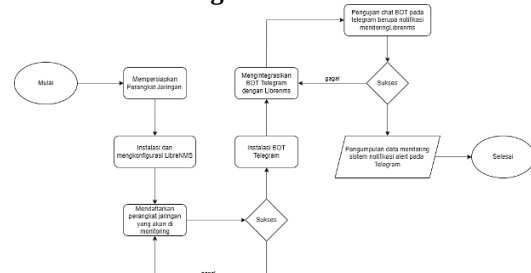
Gambar 2 menjelaskan sebuah proses pada Protokol SNMP yang berguna untuk melakukan pemantauan jaringan secara remote sehingga perangkat dapat dipantau dari jarak jauh serta memudahkan proses audit dan konfigurasi perangkat karena dapat dilakukan kapanpun dan dimanapun.

3. METODE PENELITIAN

Metodologi penelitian yang diterapkan penulis yaitu metode kualitatif, metode kualitatif bersifat deskriptif dan lebih mendalam sehingga dapat memberikan gambaran yang lebih kompleks dan mampu mencapai tujuan penelitian serta masalah yang sedang diteliti, data-data dan informasi yang lengkap juga diperlukan sebagai bahan pendukung materi pembahasan.

Dalam suatu penelitian diawali dengan penyusunan perencanaan yang tersusun secara sistematis dan logis sehingga hasil yang diberikan mampu menggambarkan kondisi yang sebenarnya. Langkah berikutnya merupakan penyusunan alur penelitian yang bertujuan untuk menghasilkan penelitian yang baik dan sejalan dengan tujuan penelitian.

3.1. Alur Perancangan Sistem



Gambar 3 Alur perancangan sistem

Alur perancangan sistem yang dibuat untuk menerapkan Alert Notification serta monitoring di sistem libreNMS untuk penjelasannya sebagai berikut:

1. Mempersiapkan network device yang akan digunakan untuk Perusahaan BLiP Integrator sebagai monitoring perangkat:
 - a) Perangkat Router dengan ip public,
 - b) Perangkat Switch dengan ip public,
 - c) Perangkat Access Point dengan ip public,
 - d) Computer server dengan Sistem Operasi Ubuntu server versi 22.04 sebagai server untuk pemasangan LibreNMS.
2. Instalasi sistem monitoring LibreNMS pada komputer server dan konfigurasi SNMP pada LibreNMS.
3. Pada sistem monitoring LibreNMS dilakukan pendaftaran IP Address perangkat yang akan dimonitoring.
4. Instalasi BOT Telegram mengikuti panduan resmi dari aplikasi Telegram untuk melakukan konfigurasi alert transport dengan cara menambahkan nama transport, type transport, chat id grup telegram, token BOT Telegram dan dilakukannya testing notification alert pada chat BOT Telegram.

3.2. Spesifikasi Perangkat

Setelah dilakukannya identifikasi, maka dilakukan analisa serta menentukan kebutuhan software dan hardware yang dibutuhkan untuk menerapkan metode yang dipakai pada notification alert system monitoring, pada tahap analisis sistem, penulis memilih komputer server sebagai mesin untuk melakukan pemantauan jaringan menggunakan LibreNMS.

Tabel 1 Spesifikasi kebutuhan perangkat

Sistem operasi	Ubuntu server 22.04
Prosesor	i5-7200U CPU 2.50GHz (4 CPUs)
RAM	4 GB
Kapasitas	HDD 250 GB

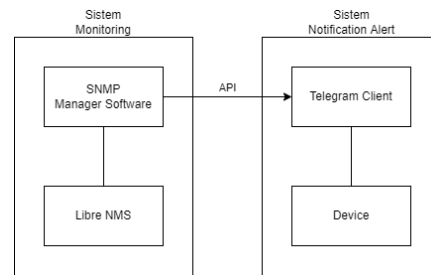
Pada Tabel 1 akan dijelaskan mengenai spesifikasi sistem operasi, prosesor, RAM, dan Kapasitas yang dipakai untuk computer server, LibreNMS akan di install pada computer server untuk sistem monitoring.

Tabel 2. Kebutuhan perangkat lunak

Perangkat Lunak	Keterangan
MariaDB	Berfungsi sebagai <i>database storage</i> pada server <i>LibreNMS</i> dalam mengelola program.
Nginx	Sebagai web server pada <i>LibreNMS</i> .
SNMP	Sebagai tool untuk memantau server <i>LibreNMS</i> .
Putty	Aplikasi ini memiliki fungsi sebagai remote akses melalui SSH pada computer server melalui jarak jauh.
Telegram	Sebagai aplikasi chatting penerima layanan transport.

Pada Tabel 2 menjelaskan beberapa kebutuhan perangkat lunak yang dipakai dalam menjalankan aplikasi LibreNMS dan menerapkan sistem monitoring.

3.3. Transport Layer



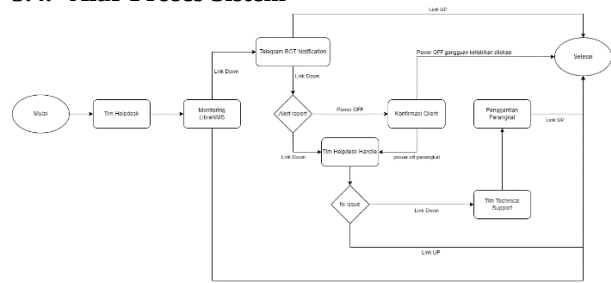
Gambar 4 Struktur proses trasport data

Transport data pada Gambar 4 merupakan suatu tahap dalam pengiriman data dari data sistem monitoring libreNMS ke sistem peringatan Bot telegram dengan menggunakan protokol atau metode tertentu, dalam proses Transport Data yang digunakan ada dua sistem utama yang akan diterapkan, yaitu Sistem Monitoring dan Sistem Notification Alert, API token yang diberikan Bot pada Telegram dapat digunakan untuk mengintegrasikan aplikasi LibreNMS dengan tujuan membuat notification alert dari LibreNMS ke grup Telegram,

Berikut merupakan langkah-langkah yang akan diterapkan pada proses transport data:

1. Melakukan Instalasi dan aktifkan fitur Telegram Notifier pada LibreNMS.
2. Konfigurasi notifikasi Telegram pada LibreNMS dengan menyalin dan memberi token BOT dan chat id pada file konfigurasi.
3. Memberikan rule untuk pengiriman notifikasi ke Telegram ChatBOT pada LibreNMS serta mengatur kondisi yang diinginkan.
4. Setelah memberikan konfigurasi dan rule notifikasi akan otomatis dikirim ke Telegram ChatBOT jika kondisi pada rule terpenuhi.

3.4. Alur Proses Sistem



Gambar 5 Alur proses sistem

Gambar 5 merupakan alur proses sistem ketika terjadi kendala terjadi pengukuran jaringan internet dengan parameter bandwidth, delay dan packet loss, serta mengukur up time – downtime dari setiap node[9].

Berikut merupakan penjelasan pada alur proses sistem :

1. Tim Helpdesk melakukan monitoring link melalui sistem monitoring libreNMS,
2. Telegram bot akan memberikan notifikasi pada aplikasi telegram ketika LibreNMS tidak dapat berkomunikasi dengan perangkat melalui udp, tcp, SNMP, alert notification akan otomatis dikirimkan ke Telegram Grup chat,
3. Pada proses alert report indikasi issue yang terjadi yaitu kendala pada power atau gangguan link, jika kendala power maka tim helpdesk akan berkoordinasi dengan pihak client, ketika link down disebabkan karena gangguan link tim helpdesk akan melakukan proses fix issue,
4. Pada proses koordinasi dengan pihak client tidak ada gangguan kelistrikan dan power pada perangkat tim helpdesk akan melakukan fix issue yang dimana gangguan tersebut merupakan gangguan link atau gangguan pada perangkat yang menyebabkan link down,
5. Proses fix issue akan menentukan hasil jika link tetap mengalami down maka dapat disimpulkan bahwa link down dikarenakan perangkat yang harus diganti dan dikoordinasikan dengan tim support, dan jika link dapat diperbaiki dari sisi tim helpdesk link dapat kembali UP.

Alur proses sistem dapat berjalan dengan lancar ketika alur perancangan sistem sudah berjalan dengan normal sesuai yang direncanakan, proses bisnis akan terjadi pada transport layer yang akan menjadi inti dari pembahasan dari penulis.

4. HASIL DAN PEMBAHASAN

Pengujian *notification alert* dapat dijalankan ketika instalasi serta konfigurasi pada libreNMS dan aplikasi Bot telegram sudah dilakukan. alur perancangan sistem dan alur proses sistem dapat menjadi dasar dari Implementasi yang dibuat oleh penulis.

4.1. Analisis

Dalam menjalankan proses pada sistem *libreNMS* dibutuhkan persiapan untuk menghubungkan perangkat yang menjadi objek dalam monitoring, beberapa konfigurasi pada *libreNMS* dan aplikasi telegram menjadi proses komunikasi.

Dalam penerapan alur perancangan sistem diperlukan persiapan-persiapan yang dibutuhkan dalam memonitoring perangkat sehingga dalam menerapkan proses transport menjadi efisien, perangkat yang terhubung pada libreNMS menggunakan SNMP dan berperan sebagai transport layer, pada tiap perangkat yang terkoneksi dengan SNMP grafik data, latency, log, spesifikasi dapat ditampilkan pada libreNMS.

pada tahap awal akan menggunakan perangkat dengan IP Public yang akan dimonitoring dalam sistem alert notification berikut 3 perangkat yang akan dipersiapkan:

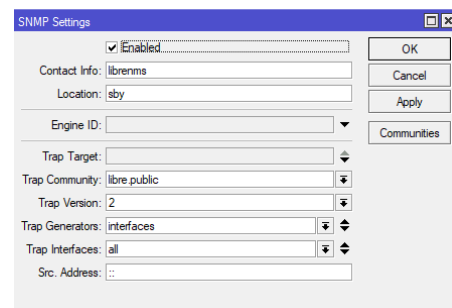
- a) IP : 119.2.48.121 sebagai devices router
- b) IP : 119.2.48.12 sebagai devices access point
- c) IP : 119.2.48.176 sebagai devices switch

Selain menentukan perangkat dan IP diperlukan konfigurasi SNMP untuk integrasi proses transport dari devices ke libreNMS lalu dimonitoring.



Gambar 6 Pembuatan SNMP communities

Dengan melakukan pembuatan communities SNMP maka perangkat dapat terkoneksi pada sistem monitoring LibreNMS.

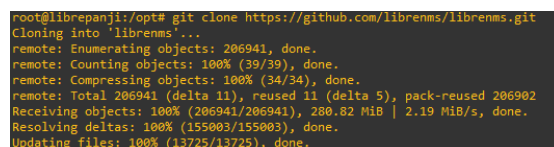


Gambar 7 Konfigurasi SNMP perangkat

Konfigurasi *SNMP* pada perangkat diharuskan sama dengan konfigurasi pada libreNMS saat melakukan add device.

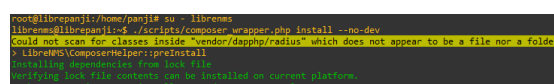
4.2. Pembahasan

Tahap instalasi yang akan dilakukan yaitu instalasi server Ubuntu lalu instalasi LibreNMS, untuk menginstall.



Gambar 8 Perintah instalasi libreNMS

Masuk sebagai root dan lakukan instalasi LibreNMS lalu tunggu hingga proses selesai. gunakan perintah *chmod* merupakan perintah dalam mengubah hak akses (permission) dari direktori libreNMS. Dan *chown*, perintah *chown* merupakan perintah untuk mengubah kepemilikan (ownership) direktori libreNMS.



Gambar 9 Perintah menginstall dependensi

Perintah tersebut digunakan untuk menginstall dependensi pada proyek instalasi *libreNMS*. untuk mengkonfigurasi php sehingga proses pembuatan,

menelola, dan memanipulasi konten web pada LibreNMS dapat berjalan.

```
root@librepanji:/home/panji# vi /etc/nginx/conf.d/librenms.conf
```

Gambar 90 Perintah konfigurasi web server nginx

Perintah pada Gambar 10 merupakan command untuk mengkonfigurasi *nginx* web server dalam menentukan cara memanggil *libreNMS* pada server ke website browser.

```
root@librepanji:/home/panji# cat /etc/nginx/conf.d/librenms.conf
server {
    listen 80;
    server_name 192.168.100.40;
    root /opt/librenms/html;
    index index.php;

    charset utf-8;

    gzip on;
    gzip_types text/css application/javascript text/javascript image/svg+xml text/plain text/xsd text/xml;
    try_files $uri $uri/ /index.php/$uri;

    location ~ [^/]\.php(/|$) {
        fastcgi_pass unix:/var/run/php-fpm.sock;
        fastcgi_split_path_info ^(.+\.php)(/.+)$;
        include fastcgi.conf;
    }

    location ~ /\.well-known {
        deny all;
    }
}
```

Gambar 101 Konfigurasi Nginx web server

Berikut seperti Gambar 11 merupakan tampilan konfigurasi *nginx* web server yang dibuat untuk instalasi *libreNMS*, konfigurasi default pada *librenms.conf* dapat dilihat pada <https://docs.librenms.org/Installation/Install-LibreNMS/> yang disediakan oleh *libreNMS* sendiri.

```
root@librepanji:~# systemctl restart nginx
root@librepanji:~# systemctl restart php8.1-fpm
```

Gambar 112 Perintah untuk memuat ulang web server

Lakukan restart setiap melakukan perubahan pada *Nginx* web server dan php lakukan perintah seperti pada Gambar 12 agar konfigurasi yang disimpan dapat dijalankan tanpa ada error.

```
root@librepanji:/home/panji# nano /etc/snmp/snmpd.conf
```

Gambar 123 Perintah konfigurasi Snmpd.conf

Perintah untuk konfigurasi snmp dilakukan dengan mengubah nama *community* pada snmp, nama *community* ini digunakan untuk menghubungkan snmp pada perangkat yang dituju.

```
root@librepanji:~# systemctl restart snmpd
root@librepanji:~# systemctl status snmpd
● snmpd.service - Simple Network Management Protocol (SNMP) Daemon.
   Loaded: loaded (/lib/systemd/system/snmpd.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-07-12 20:09:48 MIB; 1s ago
     Main PID: 8866 (snmpd)
       Tasks: 1 (limit: 48135)
        Memory: 3.0M
          CPU: 19ms
      CGroup: /system.slice/snmpd.service
              └─8866 /usr/sbin/snmpd -lOw -u Debian-snmp -g Debian-snmp -i -smux mteTrigger mteTriggerConf -f

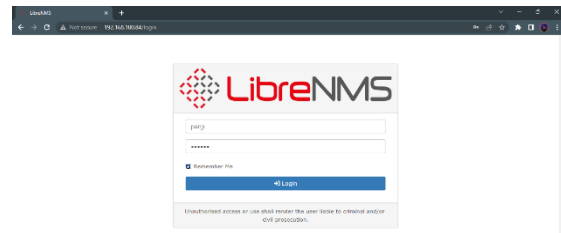
Jul 12 20:09:48 librepanji systemd[1]: snmpd.service: Deactivated successfully.
Jul 12 20:09:48 librepanji systemd[1]: Stopped Simple Network Management Protocol (SNMP) Daemon.
Jul 12 20:09:48 librepanji systemd[1]: Starting Simple Network Management Protocol (SNMP) Daemon....
Jul 12 20:09:48 librepanji systemd[1]: Started Simple Network Management Protocol (SNMP) Daemon.
```

Gambar 134 Konfigurasi SNMP pada libreNMS

Lakukan perintah restart dan cek status apakah running atau fail, service *snmpd* akan menerapkan konfigurasi yang telah diubah ketika melakukan restart dan menampilkan status.

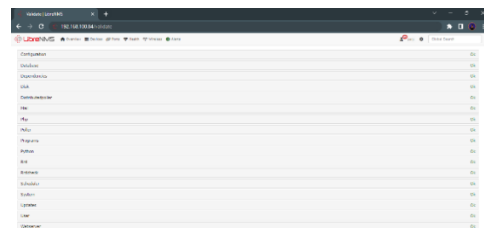
Jika *nginx* web server, php, *snmpd* sudah dikonfigurasi maka lakukan konfigurasi *libreNMS* dalam web browser, untuk mengakses website *libreNMS* dengan memasukkan IP di halaman

browser, IP yang digunakan merupakan IP yang telah dikonfigurasi pada *nginx* web server.



Gambar 14 Halaman awal login akses libreNMS

Tampilan halaman awal, diperlukan login untuk melakukan konfigurasi dan mengelola fitur dari *libreNMS*.

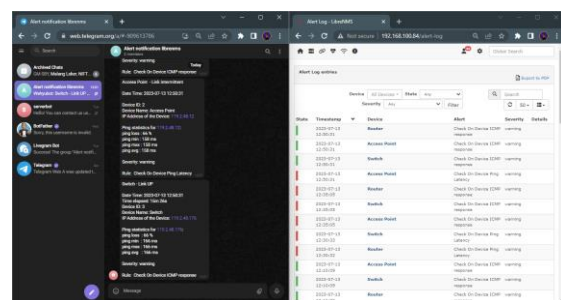


Gambar 15 Proses validasi

Proses validasi ini sangat penting untuk mengecek apakah ada konfigurasi yang gagal dijalankan, maka diperlukan status normal sehingga sistem dapat berjalan dengan lancar

4.3. Hasil Penelitian

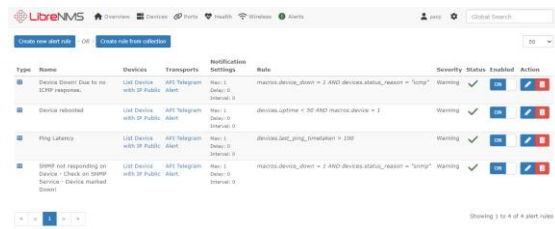
Hasil dari notifikasi status perangkat yang diteruskan oleh sistem monitoring *libreNMS* ke grup telegram menggunakan transport Bot telegram:



Gambar 16 Hasil Output

Tampilan hasil atau output dari penelitian ini, transport *LibreNMS* berhasil dilakukan pada aplikasi Telegram, untuk tampilan kiri merupakan tampilan grup pada aplikasi Telegram Web, dan kanan merupakan tampilan log pada *LibreNMS* dalam mengirimkan hasil notifikasi.

Notifikasi yang disampaikan memerlukan alert rule sebagai syarat untuk memunculkan notifikasi, dan alert template sebagai notifikasi tampilan yang akan ditampilkan oleh Bot telegram.



Gambar 17 Alert rule konfigurasi

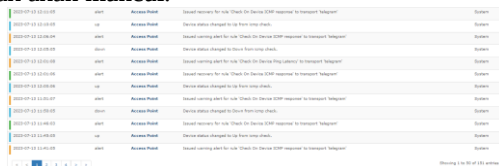
Pada Gambar 18 merupakan hasil dari konfigurasi Alert rule pada LibreNMS, pada konfigurasi ini telah disediakan template yang dapat dimodifikasi sesuai kebutuhan dan dapat dikembangkan sehingga sistem monitoring lebih optimal.

Setelah membuat alert rule, buat rule baru untuk mengubah isi rule sesuai kebutuhan, fungsi alert template yaitu untuk mengubah dan menyesuaikan isi pemberitahuan yang dikirimkan saat aturan peringatan dipicu.



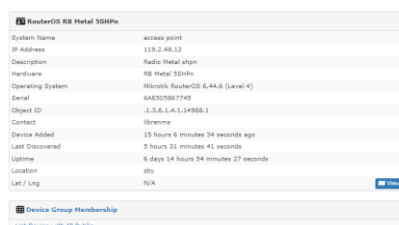
Gambar 18 Alert template

Alert template yang dibuat akan menyesuaikan Alert rule seperti Gambar 19, jika Alert rule tidak menerapkan Alert template maka sistem notifikasi tidak akan muncul.



Gambar 19 Hasil log pada device

Pada Gambar 20 merupakan tampilan hasil log pada perangkat yang dimonitoring ketika mengalami gangguan.



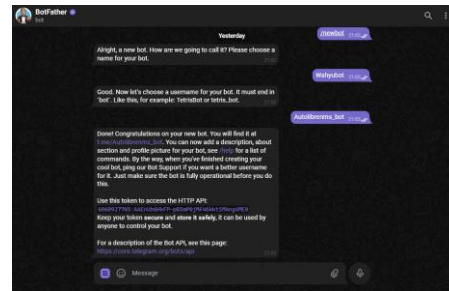
Gambar 20 Spesifikasi device

Pada Gambar 21 LibreNMS dapat menampilkan spesifikasi perangkat secara detail ketika perangkat tersebut sudah terkoneksi dengan sistem monitoring LibreNMS.

4.4. Instalasi dan Pengujian Bot Telegram

Pada instalasi BOT telegram diharuskan membuat atau memiliki aplikasi Telegram terlebih

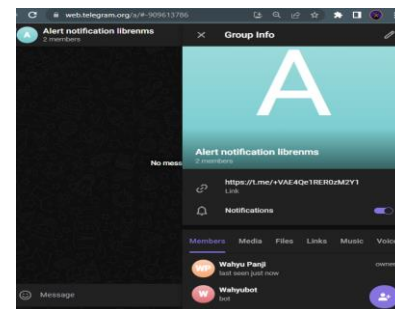
dahulu untuk memanggil BotFather, jika sudah memiliki akun telegram BotFather dapat dipanggil melalui url berikut: <https://telegram.me/botfather>,



Gambar 212 Pembuatan Bot baru pada telegram

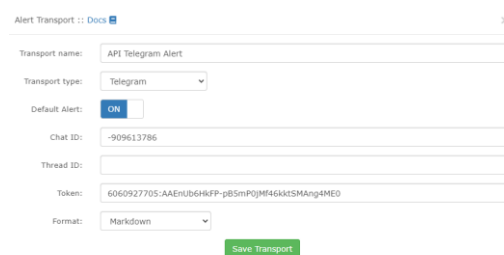
Pada Gambar 22 dijelaskan langkah singkat dalam membuat Bot pada aplikasi telegram, berikut merupakan langkah pembuatan Bot:

1. Ketikkan /newbot untuk membuat bot baru,
2. Membuat username pada Bot tersebut,
3. BotFather akan memberikan respon Done! jika username Bot berhasil dibuat, jika gagal akan muncul respon username is already taken/username is invalid, jika sudah berhasil membuat Bot akan diberikan Token akses HTTP API untuk mengintegrasikannya pada libreNMS.



Gambar 22 Membuat grup telegram untuk notifikasi

Pada Gambar 23 tampilan pada grup telegram yang sudah dibuat, Bot yang sudah dibuat dapat diundang kedalam grup tersebut untuk menjalankan sistem alert notifikasi, untuk menemukan chat ID dapat dilihat dari angka setelah hashtag[10] seperti link grup <https://web.telegram.org/a/#-909613786> yaitu [-909613786] untuk chat id yang didapat.

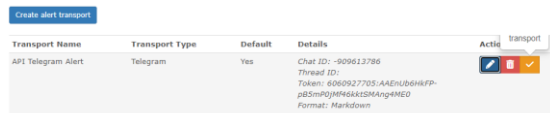


Gambar 23 Konfigurasi transport Bot telegram

Gambar 24 merupakan cara konfigurasi transport untuk proses menerapkan sistem alert notification pada

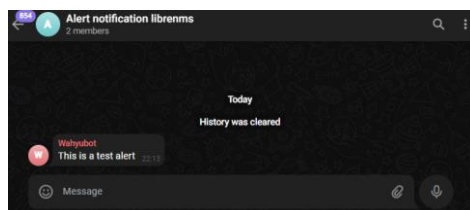
libreNMS dan aplikasi telegram, yang dibutuhkan pada konfigurasi ini yaitu :

- Transport type
- Chat Id pada telegram grup
- Token Id pada Bot telegram
- d)



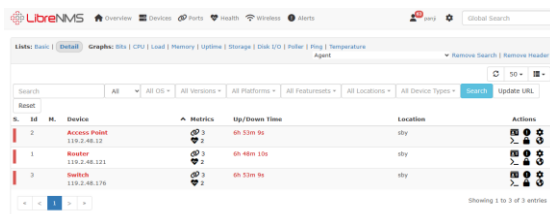
Gambar 24 Pengujian Transport dari libreNMS

Tampilan dari hasil konfigurasi yang sudah dibuat untuk pengujian dapat di coba dengan memilih transport button dengan simbol check (✓) untuk memunculkan pengujian ke grup chat telegram.

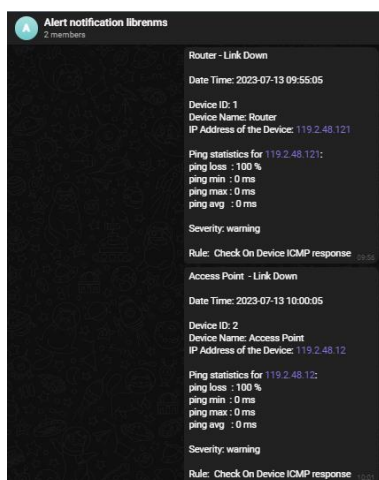


Gambar 25 Tampilan pengujian pada grup telegram

Hasil dari pengujian ini akan muncul text yang dikirimkan dari libreNMS yang akan difungsikan untuk pemantauan diluar kantor sebagai peringatan jika ada gangguan.



Gambar 26 Tampilan Down pada LibreNMS

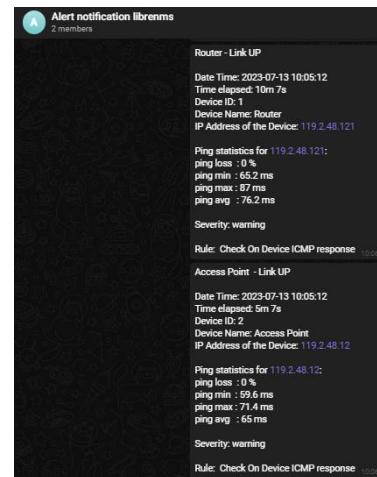


Gambar 27 Notifikasi link down

Untuk melakukan pengujian pada perangkat diperlukan down atau mematikan perangkat, pada Gambar 27 merupakan hasil tampilan perangkat

dengan status tidak aktif atau sedang mengalami gangguan, text pada libreNMS akan berubah menjadi warna merah yang dimana ICMP dan SNMP tidak merespon perangkat tersebut.

Pada Gambar 35 merupakan tampilan ketika perangkat mengalami gangguan libreNMS akan mengirimkan notifikasi kepada grup chat telegram secara real time.



Gambar 28 Notifikasi link recovery

Pada Gambar 36 merupakan tampilan ketika perangkat yang telah mengalami gangguan sudah dilakukan perbaikan maka libreNMS akan mengirimkan notifikasi berupa status perangkat kepada grup chat telegram secara real time.

5. KESIMPULAN DAN SARAN

Menerapkan Bot telegram dapat mempermudah tim technical support dan tim helpdesk dalam memperoleh informasi status jaringan dan status perangkat secara realtime, informasi kondisi fisik perangkat dari spesifikasi, grafik data, latency ping sehingga lebih efisien dan cepat mengambil keputusan dalam penanganan serta perbaikan ketika mendapat gangguan, untuk mendapatkan waktu real time dalam monitoring diperlukan sebuah Bot untuk menyampaikan informasi kedalam aplikasi telegram, proses-proses yang terjadi pada transport menggunakan ICMP dan SNMP sebagai bahan utama proses integrasi dan disempurnakan lewat alert transport sesuai ketentuan dan aturan yang dibuat.

Sebagai akhir dari kesimpulan dari penelitian ini maka dapat ditarik kesimpulan bahwa: Penerapan sistem yang digunakan sangat mempermudah tim dalam memantau kondisi jaringan untuk client/pelanggan BLiP integrator. Telah dihasilkan Sistem monitoring melalui notifikasi Bot telegram untuk memberitahu terkait gangguan jaringan secara realtime dan otomatis kepada tim helpdesk dalam memonitoring jaringan dimanapun berada. Dari hasil monitoring jaringan gangguan yang sering dialami akan cepat terselesaikan dan diprogress ketika perangkat mengalami gangguan dan cepat terprogress supaya downtime link tidak akan terjadi

terlalu lama yang menyebabkan penurunan SLA pada perusahaan. Dari hasil pengujian Bot telegram terhadap alert rule dapat difungsikan untuk memberikan berbagai aturan dalam memonitor jaringan dalam sistem librenms.

Setiap hasil dari penelitian yang telah dilakukan para penulis, selalu memiliki beberapa saran untuk digunakan dalam penelitian selanjutnya yaitu: Membuat sistem alert notification lain menggunakan aplikasi Instant Messaging lain cadangan apabila terdapat masalah pada Bot yang sedang berjalan, Membuat Analisa pada sistem monitoring dan notifikasi berkelanjutan untuk dapat memperoleh hasil yang praktis dan efektif. Penerapan Bot telegram pada libreNMS ini tidak hanya untuk bidang ICT dan penyedia layanan-layanan internet namun masih bisa dikembangkan pada berbagai bidang. Sistem Monitoring berbasis web libreNMS dapat dikembangkan lagi selain menggunakan API masih ada ide-ide yang dapat diterapkan, penelitian ini masih jauh dari kata sempurna diperlukan pengembang berikutnya secara berkelanjutan agar mampu melakukan konfigurasi, implementasi lain serta mampu menghasilkan notifikasi yang lebih detail.

DAFTAR PUSTAKA

- [1] J. Simangunsong and A. Voutama, "RANCANG BANGUN SISTEM INFORMASI ONLINE MARKETPLACE BERBASIS WEB APPLICATION STUDI KASUS: KLG CAMPUS RESIDENCE," vol. 7, no. 2, pp. 1261–1268, 2023.
- [2] H. R. Tasmara and A. D. Indriyanti, "Analisis Grade of Services dengan Erlang Theory," vol. 03, no. 04, pp. 112–118, 2022.
- [3] W. Adhiwibowo, F. Wahyu Christanto, and A. Firman Daru, "Implementasi API Bot Telegram untuk Sistem Notifikasi pada The Dude Network Monitoring System," pp. 593–599, 2021.
- [4] Rakhmat Dwi Jayanto, "Rancang Bangun Sistem Monitoring Jaringan Menggunakan Mikrotik Router OS," *JATI (Jurnal Mhs. Tek. Inform.,* vol. 3, no. 4, pp. 391–395, 2019.
- [5] H. Kuswanto, "Sistem Monitoring Perangkat Jaringan Menggunakan Protokol SNMP Dengan Notifikasi Email," *J. Tek. Komput.,* vol. 4, no. 2, pp. 99–104, 2018, doi: 10.31294/jtk.v4i2.3447.
- [6] A. Cokrojoyo, J. Andjarwirawan, and P. S. T. I. F. <http://studentjournal.petra.ac.id/index.php/teknik-informatika/article/view/5163>Agustinu. Noertjahyana, Cokrojoyo, A., Andjarwirawan, J., & Noertjahyana, A. (2017). Pembuatan Bot Telegram Untuk Mengambil Informasi dan Jadwal Film Menggunakan PHP. *Jurnal Infra*, 5(1), 224–227, "Pembuatan Bot Telegram Untuk Mengambil Informasi dan Jadwal Film Menggunakan PHP," *J. Infra*, vol. 5, no. 1, pp. 224–227, Program Studi Teknik Informatika Fakultas, 2017, [Online]. Available: <http://studentjournal.petra.ac.id/index.php/teknik-informatika/article/view/5163>
- [7] M. Efendy and M. Achlaq Mizanul, "Implementasi Sistem Monitoring Dan Backup Konfigurasi Perangkat Jaringan Menggunakan Librenms Di Pt. Data Utama Dinamika," *Smart Comp Jurnalnya Orang Pint. Komput.,* vol. 11, no. 4, pp. 668–673, 2022, doi: 10.30591/smartcomp.v11i4.4253.
- [8] S. Arifin, "Implementasi Monitoring Jaringan Menggunakan Raspberry Pi Dengan Memanfaatkan Protokol Smt (Simple Mail Transfer Protocol)," *J. Mhs. Tek. Inform.,* vol. 1, no. 1, pp. 173–179, 2017.
- [9] A. Tenriawaru, S. Subardin, and N. Nurkaeani, "Pengembangan Sistem Monitoring Jaringan Komputer Menggunakan Dude," *Digit. Transform. Technol.,* vol. 2, no. 2, pp. 21–26, 2022, doi: 10.47709/digitech.v2i2.1780.
- [10] N. R. Fachrurrozi, A. A. Wirabudi, and 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. 111–122, 2023, doi: 10.22441/sinergi.2023.1.013.