

Analisis Grade of Services dengan Erlang Theory

Hasbi Rivanda Tasmara¹, Aries Dwi Indriyanti²

^{1,2} Jurusan Teknik Informatika/Program Studi S1 Sistem Informasi, Universitas Negeri Surabaya

¹hasbi.18051@mhs.unesa.ac.id

²ariesdwi@unesa.ac.id

Abstrak— Dalam perkembangan teknologi saat ini, jaringan selalu dibutuhkan dalam memenuhi kebutuhan *transfer data* pada setiap perangkat yang terkoneksi. Terutama pada saat pandemi *covid-19* hampir segala kegiatan dilakukan secara daring menyebabkan peningkatan penggunaan jaringan oleh pengguna internet. Salah satu hal yang dapat dilakukan adalah menerapkan *Network Monitoring System* yang bertujuan untuk memantau lalu lintas pada perangkat jaringan sudah baik ataupun sebaliknya dengan bantuan aplikasi *LibreNMS*. *LibreNMS* merupakan aplikasi *monitoring* jaringan yang bersifat *open source* dan memiliki banyak fitur. Misalnya, secara otomatis dapat memindai perangkat jaringan, memantau status penggunaan dan lalu lintas data, serta memantau status perangkat jaringan yang diwakili oleh *Rrdtool* sebagai alat produksi grafis. Hasil implementasi sistem *monitoring* menggunakan *LibreNMS* dapat melakukan pemantauan kondisi trafik data serta pemantauan kondisi *resource*, dan juga mampu melakukan sistem peringatan pada perangkat yang memudahkan administrator untuk memantau jaringan. Kualitas jaringan juga dapat diketahui dengan melakukan penerapan teori pada data yang didapat dalam *LibreNMS*. Penerapan teori ini menggunakan teori *Erlang* yang dapat mengetahui *Grade of Services* pada kualitas jaringan yang ada.

Kata Kunci— *Network Monitoring System, LibreNMS, SNMP, Erlang Theory*.

I. PENDAHULUAN

Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), Menunjukkan bahwa survei tingkat penetrasi pengguna internet Indonesia mengalami peningkatan pada tahun 2018[1]. Menurut hasil survei tersebut, jumlah pengguna internet meningkat menjadi 171,17 juta pada tahun 2018. Penggunaan internet di Indonesia pada tahun 2019 – 2020 (Q2) mencapai 196.71 juta jiwa dari total populasi penduduk Indonesia yang berjumlah 266.91 juta jiwa, dari angka tersebut tentunya penggunaan internet di Indonesia akan semakin bertambah mengingat bahwa jaringan selalu dibutuhkan dalam memenuhi kebutuhan transfer data dan sinkronisasi dari setiap perangkat yang dipakai guna menunjang infrastruktur yang ada.

Dengan adanya peningkatan penggunaan internet, kinerja sumber daya perangkat jaringan perlu ditingkatkan, dan pengawasan perlu diperkuat untuk memastikan stabilitas bisnis [2]. Oleh karena itu, diperlukan suatu sistem untuk memonitoring peralatan jaringan[3]. *Network Monitoring System* (NMS) adalah sistem yang digunakan untuk memonitor suatu jaringan dan sebuah perangkat seperti *server, router, switch*, dan *PC* dengan memasukkan *ip address*[4].

Pada PT Radnet Digital Indonesia saat ini diberlakukan penerapan WFH (*Work From Home*) yang mengharuskan Admin *Server* bekerja di rumah, namun sistem *monitoring* PT Radnet Digital Indonesia masih memakai *IP Address* lokal. Oleh karena itu sebaiknya *monitoring* jaringan menggunakan

IP Address Public supaya dapat dilakukan *monitoring* secara online[5].

Saat ini banyak aplikasi yang dapat digunakan sebagai pemantau perangkat jaringan seperti *LibreNMS*[6]. *LibreNMS* merupakan aplikasi *open source* berbasis *PHP*. *LibreNMS* menggunakan *protocol SNMP* yang berfungsi untuk memonitor beberapa perangkat jaringan[7]. Pada *LibreNMS* juga sudah mendukung *module transport* yang berguna untuk mengambil data dari sistem dan menampilkannya dalam bentuk grafik[8]. Aplikasi *LibreNMS* bersifat *agnostic*, artinya aplikasi *LibreNMS* dapat dijalankan pada berbagai macam tipe Sistem Operasi yang tersedia[9]. *LibreNMS* juga memiliki keunggulan pada fitur-fiturnya seperti fitur *auto discovery* yang berfungsi untuk mengumpulkan semua informasi tentang perangkat yang menggunakan beberapa kontrol (seperti *OSPF, BGP, SNMP*, dan lainnya), *alerting* atau sistem peringatan yang dapat membantu pengguna mengetahui permasalahan dengan beberapa metode (via *email, telegram, slack*, dan lain-lain), *Service Monitoring* yang dapat membantu pengguna dalam melihat trafik penggunaan dan memantau *host resource, Device Backup Integration* yang mendukung sebuah sistem untuk menjadwalkan backup secara berkala.

Monitoring jaringan berfungsi untuk mengetahui penyebab sistem jaringan komputer saat terjadi kendala, kendala jaringan komputer terdiri dari banyak faktor salah satunya adalah terputus dari penyedia layanan internet atau (ISP)[10]. *Monitoring* jaringan menggunakan *LibreNMS* dapat mempermudah admin jaringan dalam menemukan permasalahan yang ada sehingga dapat lebih cepat dalam melakukan perbaikan sistem[11].

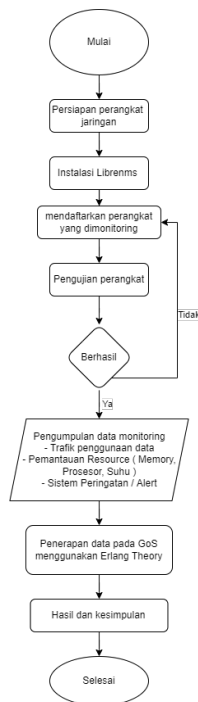
Grade of Services (GOS) adalah metode yang digunakan untuk mengetahui kualitas jaringan seperti terjadinya bloking, besar trafik, dan jumlah link, *Grade Of Services* (GOS) sendiri berada pada model trafik *Erlang-B*. *Erlang* adalah satuan intensitas trafik yang diambil dari nama seorang ilmuwan Denmark, Agner Krarup Erlang (1878-1929).

Berdasarkan latar belakang tersebut, penulis bertujuan untuk melakukan penelitian dalam bentuk jurnal ilmiah yang berjudul “Analisis Grade of Services dengan *Erlang Theory*”.

II. METODE PENELITIAN

Penelitian ini merupakan penelitian kuantitatif. Dalam penelitian ini data yang diperoleh melalui eksperimen yang nantinya akan diterapkan dalam *Erlang Theory* dan diolah dalam bentuk angka.

A. Analisa Sistem



Gbr. 1 Alur Kerja

Flowchart berikut merupakan alur kerja pada sistem monitoring yang akan dibuat beserta penjelasan sebagai berikut:

1. Menyiapkan beberapa perangkat jaringan seperti *Router*, *Switch*, *Access Point*, dan Sistem Operasi Linux yang digunakan sebagai *server private* pada *LibreNMS*, serta Sistem Operasi Windows sebagai *client* untuk mengakses *LibreNMS*.
2. Melakukan instalasi *LibreNMS* pada *Virtual Private Server* dan konfigurasi *SNMP*.
3. Melakukan pendaftaran alamat IP pada perangkat yang akan dimonitoring pada *LibreNMS*.
4. Uji coba koneksi pada perangkat yang akan dimonitoring.
5. Mengidentifikasi dan mengumpulkan data pemantauan jaringan.
6. Mengolah data menggunakan *Erlang Theory* dalam mencari *Grade of Services* pada jaringan.

Terdapat beberapa penjelasan mengenai perangkat yang digunakan, antara lain sebagai berikut:

1. *Server VPS* dengan alamat IP 103.150.151.38 yang didalamnya terdapat Aplikasi *LibreNMS* yang memantau setiap perangkat jaringan yang terhubung dengan *SNMP Agent*.
2. *Server Buaya* adalah *server mirror* dengan alamat IP 202.154.57.167 sebagai perangkat *SNMP Agent*.
3. *Server Klas* adalah *website server client* yang ada pada PT Radnet Digital Indonesia dengan alamat IP 202.154.58.26 sebagai perangkat *SNMP Agent*.

4. *Mikrotik Gateway* adalah perangkat yang ada pada PT Radnet Digital Indonesia

B. Spesifikasi Kebutuhan perangkat

Spesifikasi komputer yang digunakan untuk melakukan pemantauan jaringan menggunakan *LibreNMS* dapat dijelaskan pada Tabel I

TABLE I
SPESIFIKASI KEBUTUHAN PERANGKAT

Sistem Operasi	Ubuntu Desktop 20.04 LTS 64bit
Processor	1 Core
RAM	4GB

Terdapat beberapa perangkat lunak yang dibutuhkan untuk menjalankan aplikasi *LibreNMS* dapat dijelaskan pada Tabel II berikut

TABLE II
SPESIFIKASI PERANGKAT LUNAK

Perangkat Lunak	Keterangan
Putty	Aplikasi yang berfungsi sebagai <i>remote akses</i> melalui <i>SSH</i> pada <i>Virtual Private Server</i>

Penelitian ini berfokus untuk mengetahui kualitas jaringan menggunakan *GoS*, dengan menghitung besaran penggunaan *bandwith* dengan metode perhitungan standar *Erlang* dalam *Network Monitoring* pada *LibreNMS* menggunakan *MRTG* berbasis *SNMP*. *Bandwith* merupakan cakupan frekuensi yang digunakan oleh sinyal dalam medium transmisi secara luas[12]. Secara umum, lalu lintas adalah pergerakan suatu benda dari satu tempat ke tempat lain. dalam telekomunikasi, yang dimaksud dengan hal-hal adalah sinyal informasi seperti pulsa dan *frekuensi*. Jadi lalu lintas adalah *transfer* sinyal informasi dari satu tempat ke tempat lain melalui media telekomunikasi. Jumlah lalu lintas adalah objek pengukuran lalu lintas, jumlah pekerjaan pada suatu peralatan / saluran diukur berdasarkan waktu (kapan dan untuk berapa lama).

TABLE III
GoS STANDART ERLANG THEORY

Category	Grade
Bad	>4%
Normal	3%
Good	1-2%
Best	<1%

C. Teori Erlang-B

Erlang menentukan tingkat layanan (*Grade of Service*) pada sistem panggilan yang hilang (*Lost-Coll System*) yang memiliki saluran N dengan lalu lintas yang ditawarkan. *Pure-chance*

Traffic, yaitu kedatangan panggilan pada suatu sistem adalah peristiwa acak dan bersifat independen. Kesetimbangan statistik, yaitu probabilitas lalu lintas pada suatu sistem yang tidak berubah atau sistem dengan proses yang tidak terduga dan stasioner. Ketersediaan Penuh adalah setiap panggilan masuk diteruskan ke setiap saluran keluar. Dan panggilan masuk selama waktu sibuk akan hilang. Probabilitas panggilan yang hilang adalah distribusi *Erlang* pertama dan disebut *GoS* atau *Erlang-B* seperti yang ditunjukkan dalam persamaan (1). Persamaan diberikan simbol $E1$, $N(A)$, yang mewakili probabilitas kehilangan panggilan pada jumlah saluran N . Ketersediaan Penuh dengan lalu lintas yang ditawarkan dari *An Erlang* dapat dihitung dengan Persamaan (1)[13]. Untuk memudahkan menemukan perhitungan nilai *GoS* dalam sebuah program komputer, dapat digunakan dengan melakukan iterasi seperti pada Persamaan (2). Persamaan adalah Persamaan *Erlang-B*[14].

$$B = E_{1,N(A)} \frac{A^N / N!}{\sum_{K=0}^N A^K / K!} \quad (1)$$

$$B = E_{1,N(A)} \frac{A E_{1,N-1}(A)}{N + A E_{1,N-1}(A)} \quad (2)$$

D. Traffic Calculation

MRTG memberikan hasil dari total penggunaan *bandwidth* data dalam *bit* selama periode pengamatan. *Volume* lalu lintas dalam jam dapat ditentukan dengan membagi *bandwidth* tak terbatas dalam periode pengamatan dengan *bit rate* rata-rata untuk setiap saluran. seperti yang terlihat dalam Persamaan (3)[13]. Nilai Intensitas Lalu Lintas diperoleh dalam periode pengamatan menggunakan Persamaan (4)[13]. Nilai *GoS* dapat diambil pada jam tersibuk di jaringan penyedia. Persamaan 1 dapat digunakan untuk menemukan *GoS* dengan memasukkan nilai Intensitas Lalu Lintas dari persamaan (3).

$$\text{Traffic Volume} = (\text{Bandwith}) / (\text{Bit Rate}) \quad (3)$$

$$\text{Traffic Intensity} = (\text{Volume}) / (\text{Observation Period}) \quad (4)$$

III. HASIL DAN PEMBAHASAN

Pada tahap ini membahas hasil dari instalasi, konfigurasi Aplikasi *librenms* pada *server IP Public* dan hasil pemantauan dalam perangkat jaringan serta perhitungan *Quality of service* menggunakan *erlang theory*.

A. Konfigurasi librenms

Pada implementasi ini akan dilakukan instalasi pada *server Ubuntu* dan melakukan instalasi *LibreNMS* dengan memasukkan perintah berikut:

```
$ sudo git clone https://github.com/librenms/librenms.git
```

Gbr. 2 Perintah instalasi librenms

Untuk memastikan *LibreNMS* telah di *clone* pada *server* dapat dilihat dengan memasukkan perintah *ls* di file “*opt/librenms*”:

```
$ sudo ls /opt/librenms/
```

Gbr. 3 Perintah pengecekan instalasi librenms

Agar *LibreNMS* dapat di akses tanpa *root* maka harus memasukkan perintah berikut :

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

Gbr. 4 Konfigurasi untuk mengubah hak akses pada librenms

Sebelum menjalankan *LibreNMS*, perlu mengatur *php* terlebih dahulu sebagai tempat berjalannya *LibreNMS*, dengan mengetikkan perintah berikut:

```
cp /etc/php/7.4/fpm/pool.d/www.conf /etc/php/7.4/fpm/pool.d/librenms.conf
nano /etc/php/7.4/fpm/pool.d/librenms.conf
```

Gbr. 5 Perintah konfigurasi PHP

Setelah melakukan konfigurasi pada *php*, maka akan dilakukan konfigurasi pada *web server*, dengan memasukkan perintah sebagai berikut :

```
nano /etc/nginx/conf.d/librenms.conf
```

Gbr. 6 Perintah untuk mengkonfigurasi web server

Setelah memasukkan perintah tersebut, maka akan ditambahkan konfigurasi didalamnya sebagai berikut :

```
server {
    listen 80;
    server_name 103.150.151.38;
    root /opt/librenms/html;
    index index.php;

    charset utf-8;
    gzip on;
    gzip_types text/css application/javascript text/javascript application/x-javascript image/svg+xml;
    try_files $uri $uri/ /index.php?$query_string;

    location ~ [^/]\.php(/|$) {
        fastcgi_pass unix:/run/php-fpm-librenms.sock;
        fastcgi_split_path_info ^(.+\.php)(/.+)$;
        include fastcgi.conf;
    }
    location ~ /\. (?!(well-known)).* {
        deny all;
    }
}
```

Gbr. 7 Konfigurasi pada web Server

Setelah dilakukan konfigurasi pada *web server*, maka kita *restart* terlebih dahulu supaya konfigurasi yang disimpan dapat dijalankan dengan memasukkan perintah berikut :

```
systemctl restart nginx
systemctl restart php7.4-fpm
```

Gbr. 8 Perintah untuk memuat ulang web server

Setelah melakukan *restart web server* dan *php*, maka dilakukan konfigurasi pada *file snmpd.conf* sebagai berikut:

```
/etc/snmp/snmpd.conf
```

Gbr. 9 perintah untuk mengkonfigurasi SNMP

Pada konfigurasi *snmpd.conf* dilakukan penambahan nama *community* yang akan digunakan pada *LibreNMS*.

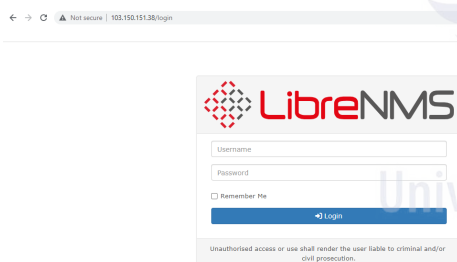
Setelah dilakukan konfigurasi, *service snmpd* akan dijalankan ulang untuk menerapkan konfigurasi yang sudah disimpan dan melihat apakah *service* sudah berjalan dengan perintah sebagai berikut:

```
root@monitoring:~# systemctl restart snmpd
root@monitoring:~# 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 2022-06-08 05:09:12 UTC; 5s ago
     Process: 942776 ExecStartPre=/bin/mkdir -p /var/run/agentx (code=exited, status=0/SUCCESS)
    Main PID: 942777 (snmpd)
       Tasks: 1 (limit: 2274)
      Memory: 7.4M
    CGroup: /system.slice/snmpd.service
            └─942777 /usr/sbin/snmpd -Low -u Debian-snmp -g Debian-snmp -I -smux mteTrigger #
Jun 08 05:09:12 monitoring systemd[1]: Starting Simple Network Management Protocol (SNMP) Daemon.
Jun 08 05:09:12 monitoring systemd[1]: Started Simple Network Management Protocol (SNMP) Daemon.
lines 1-12/12 (END)
```

Gbr. 10 Konfigurasi SNMP pada librenms

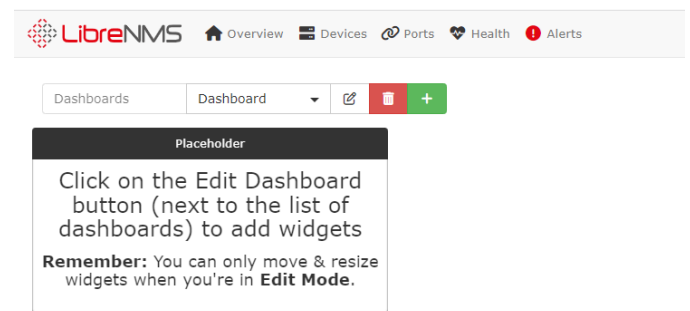
Setelah semua sudah dikonfigurasi maka akan dilakukan konfigurasi melalui *web* untuk menyempurnakan settingan *LibreNMS*, dengan memasukkan alamat *IP* di halaman *browser*.

Kredensial untuk mengakses *website LibreNMS* berasal pada *file .env* yang telah dibuat. Setelah memasukkan *username* dan *password*, maka akan dialihkan ke halaman *dashboard*.



Gbr. 11 Halaman Login LibreNMS

Pada halaman *dashboard*, penambahan perangkat jaringan dapat dilakukan pada menu *device – add device*, seperti pada Gambar 13.



Gbr. 12 Halaman Dashboard LibreNMS

Add Device

Devices will be checked for Ping/SNMP reachability before being probed.

Hostname or IP:

SNMP: ☒

SNMP Version: Port: UDP:

Port Association Mode:

SNMPv3 Configuration

Auth Level:

Auth User Name:

Auth Password:

Auth Algorithm:

Crypto Password:

Crypto Algorithm:

Force add (No ICMP or SNMP checks performed): ☐

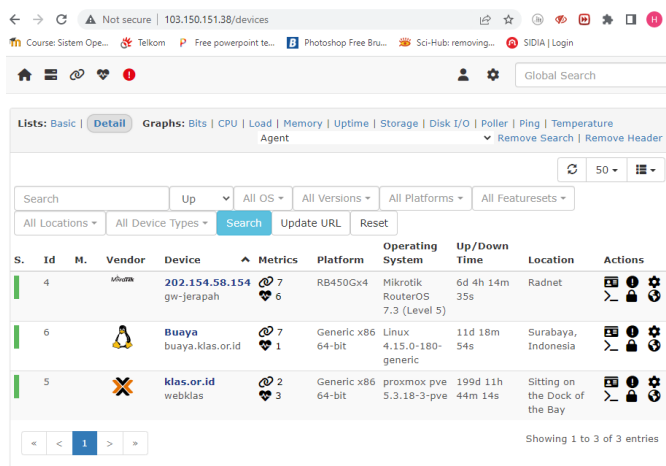
Add Device

Gbr. 13 Halaman penambahan perangkat pada LibreNMS

Penambahan perangkat dilakukan dengan cara memasukkan alamat *IP* dan nama *community*. Nama *community* berasal dari konfigurasi *SNMP* pada perangkat yang akan dimonitoring. Seperti pada Gambar 14, perangkat tersebut dikonfigurasi dengan *SNMP* versi 3.

B. Hasil Konfigurasi Librenms

Setelah mengkonfigurasi *SNMP* pada perangkat jaringan, dapat dilihat pada Gambar 15 bahwa perangkat berhasil dimasukkan ke dalam sistem, terdapat beberapa perangkat jaringan diantaranya yaitu *Router* dan *Server*. Di dalamnya dilengkapi detail pada perangkat jaringan diantaranya terdapat lokasi, nama sistem operasi, *Up/Down time*, jumlah *port* yang ada pada setiap perangkat jaringan serta terdapat fitur aksi yang dapat melakukan interaksi berupa *ssh*, *telnet*, *http*, *ping*, dan *ftp* terhadap *client* atau perangkat jaringan yang terdaftar.

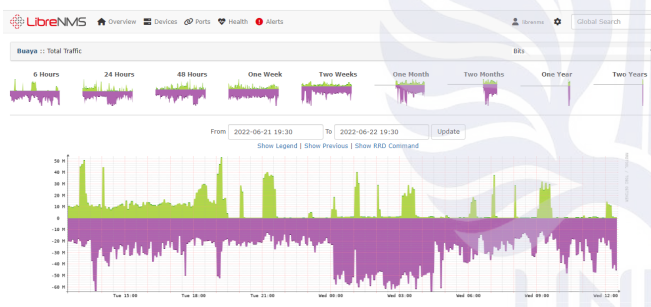


S.	ID	M.	Vendor	Device	Metrics	Platform	Operating System	Up/Down Time	Location	Actions
4	202.154.58.154	gw-jerapah	u...		7 6	RB450Gx4	Mikrotik RouterOS 7.3 (Level 5)	6d 4h 14m 35s	Radnet	
6	buaya	buaya.klas.or.id			7 1	Generic x86 64-bit	Linux 4.15.0-180-generic	11d 18m 54s	Surabaya, Indonesia	
5	klas.or.id	webklas			2 3	Generic x86 64-bit	proxmox pve 5.18-3-pve	199d 11h 44m 14s	Sitting on the Dock of the Bay	

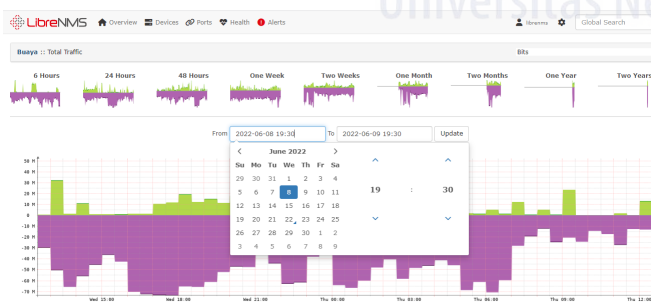
Gbr. 14 Halaman perangkat jaringan yang terdaftar

C. Hasil Pengumpulan Data Monitoring

Pada perangkat jaringan yang terdaftar, LibreNMS mampu melakukan pemantauan trafik data, CPU, dan Memory disajikan dalam bentuk grafik. Hasil pemantauan trafik data dapat dilihat dalam jangkauan setiap 6 jam, 24 jam, bahkan hingga tahunan seperti Gambar 16. Serta dapat mengkustomisasi pemantauan sesuai tanggal dan waktu yang ditentukan seperti pada Gambar 17.



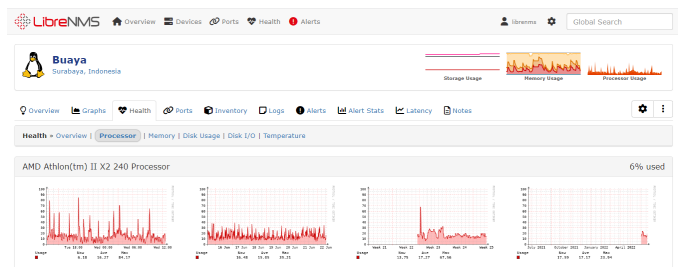
Gbr. 15 Grafik data



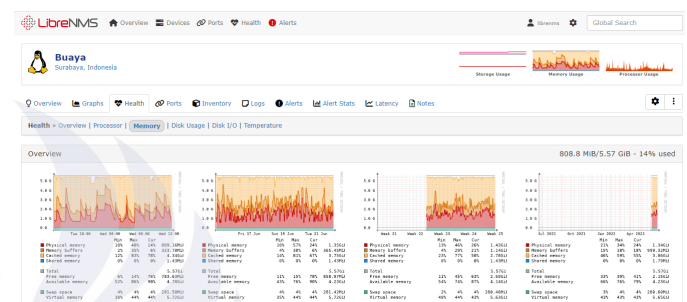
Gbr. 16 Kustomisasi tanggal pemantauan data

Selain pemantauan *traffic data*, LibreNMS juga dapat melakukan pemantauan kondisi pada perangkat jaringan seperti

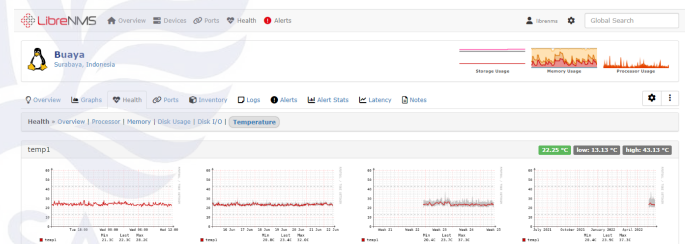
pemantauan CPU seperti Gambar 18. Pada Gambar 19 adalah tampilan pemantauan Memory serta untuk memonitoring suhu pada perangkat dapat dilihat seperti pada Gambar 20.



Gbr. 17 Grafik pemantauan CPU



Gbr. 18 Grafik Pemantauan Memory



Gbr. 19 Grafik Pemantauan Suhu

D. Analisis penggunaan Bandwith

Untuk mengetahui karakteristik pengguna pada jaringan secara detail, analisis penggunaan *bandwidth* dibagi menjadi beberapa rentang waktu yaitu hari, minggu dan bulan[15]. Disini penulis menggunakan rentang waktu harian sebagai acuan untuk menghitung penggunaan data trafik internet. Pengujian analisis *bandwith* membutuhkan beberapa faktor sebagai asumsi dasar pada trafik jaringan, dari hasil grafik yang sudah ada maka dapat disimpulkan sebagai berikut:

1. Router Kantor Radnet
Total bit rate : 100Mbps
= 102400kbps

Jumlah Host pada router : 30

Bandwidth rata-rata per kanal: $\frac{102400}{30} = 4143,3 \text{ Kbps}$

2. Server Mirror Buaya.klas.or.id

Total bit rate : 1GBbps
= 1024000Kbps

Jumlah Host pada : 121925

Server mirror

Bandwidth rata-rata per kanal: $\frac{1024000}{121925} = 8,39 \text{ Kbps}$

3. Server website Buaya.klas

Total bit rate : 300Mbps
= 307200Kbps

Jumlah Host pada : 64

Server mirror

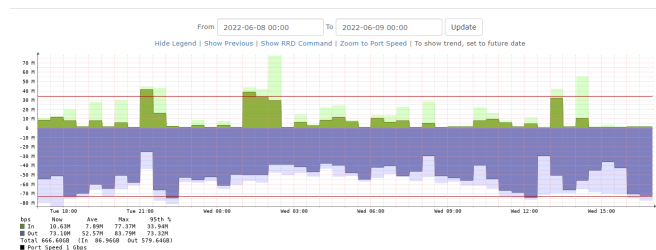
Bandwidth rata-rata per kanal: $\frac{307200}{64} = 4800 \text{ Kbps}$

Setelah data dari penggunaan bandwidth didapatkan, maka akan didapatkan *traffic intensity* seperti table IV.

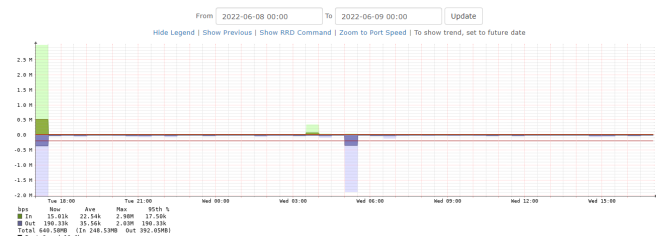
TABLE IV
TRAFFIC INTENSITY

Perangkat	Tanggal	Bandwidth	Traffic Volume (in / out)	Traffic intensity (Erlang)	Lama Pengamatan (jam tersibuk)
Router Mikrotik	08/06/22	100 Mb	0,195089	0.0975 Erlang	2 jam
			0.147583	0.0737 Erlang	2 jam
Buaya.klas.or.id	08/06/22	1000Mb	14950,76	622.9 Erlang	24 jam
			164848,02	6868.6 Erlang	24 jam
Website Buaya	08/06/22	300 Mb	0,014726	0.0073 Erlang	2 jam
			0,09709	0.0485 Erlang	2 jam

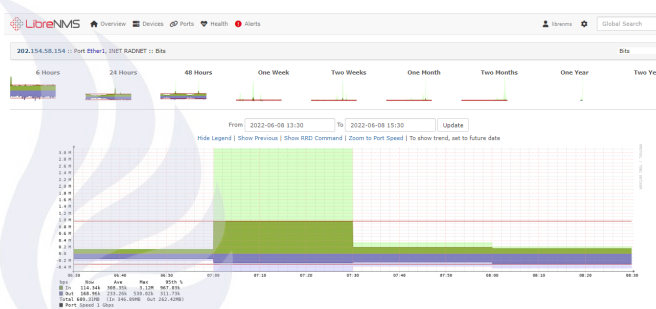
Grade of Service dilakukan untuk mengetahui apakah kinerja *server/router* pada jaringan luar pada Radnet Digital Indonesia. Di bawah ini adalah grafik pada jam tersibuk.



Gbr. 20 Monitoring jam tersibuk pada server Buaya



Gbr. 21 Monitoring jam tersibuk pada server Klas



Gbr. 22 Monitoring jam tersibuk pada Router

Dari perhitungan diatas maka berikut tabel hasil dari *GoS* pada setiap koneksi.

TABLE V
GRADE OF SERVICES

Perangkat	Koneksi	Grade of Services
Mikrotik	In bound	1.6000541628
	Out bound	3.7013236101
Buaya	In bound	0
	Out bound	0
Web Klas	In bound	2.42601020226
	Out bound	6.14820902574

Dari data diatas maka dapat disimpulkan *Grade of Services* di perangkat *mikrotik* baik pada *in bound* dan *out bound* adalah normal, *Grade of Services* di *server buaya* sangatlah baik pada *in bound* beserta *out bound*, serta *Grade of Services* di *Website Klas* adalah baik pada *in bound* dan *out bound* adalah buruk.

IV. KESIMPULAN

Kesimpulan yang dapat diambil dari Analisis Grade of Services dengan *Erlang Theory*:

1. Implementasi Sistem Monitoring Jaringan ini dibuat untuk memudahkan dalam memantau kondisi koneksi jaringan untuk client PT Radet Digital Indonesia
2. Pada software *LibreNMS* dapat diketahui status ketersediaan (*availability*) dan besarnya lalu lintas data pada perangkat jaringan (*traffic*).
3. Sesuai dengan tabel *Grade of Services* pada perangkat mikrotik terhitung baik, dan pada server buaya sangatlah baik, namun pada *website* kelas masih buruk.

V. SARAN

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat digunakan dalam penelitian selanjutnya yaitu:

1. Menambahkan Sistem Monitoring Jaringan lain sebagai cadangan apabila ada masalah pada Sistem Monitoring Jaringan yang berjalan, juga untuk lebih melengkapi fungsi monitoring pada aspek lain yang belum ada pada *LibreNMS*.
2. Melakukan Analisa secara berkelanjutan untuk mendapatkan hasil yang efektif
3. Sistem Monitoring Jaringan *LibreNMS* dapat dikembangkan dengan *API* yang tersedia guna meningkatkan efektifitas dalam memonitor jaringan supaya lebih baik.

REFERENSI

- [1] "Asosiasi Penyelenggara Jasa Internet Indonesia." <https://apjii.or.id/content/read/104/348/BULETINAPJII-EDISI-22--Maret-2018> (accessed Jul. 08, 2022).
- [2] M. R. Pratama, R. Munadi, and Hafidudin, "Implementasi Dan Analisis Sistem Monitoring Menggunakan Simple Network Management Protocol (snmp) Pada Gedung A, n.o Di Jaringan Telkom University," *e-Proceedings Eng.*, vol. 4, pp. 2092–2099, Aug. 2017, Accessed: Jul. 08, 2022. [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/en>
- [3] S. P. Rahayu and I. G. L. P. E. Prisma, "Implementasi Monitoring Manajemen Jaringan Dengan Software The Dude Berbasis Telegram Messenger," *J. Informatics Comput. Sci.*, 2022.
- [4] Y. H. Nugroho, N. P. Sastra, and D. M. Wiharta, "Analisis Unjuk Kerja Pemantauan Jaringan OpenNMS (Open Network Monitoring System) pada Jaringan TCP/IP," *J. SPEKTRUM*, vol. 5, no. 2, pp. 158–166, 2018.
- [5] D. O. Pradana, "Implementasi Notifikasi Menggunakan Telegram Messenger Pada Software The Dude Network Monitoring," *J. Manaj. Inform.*, vol. 11, no. 1, 2020.
- [6] LibreNMS, "Community-based gpl-licensed network monitoring system," *Librenms*. <http://www.librenms.org/>. (accessed Jul. 10, 2022).
- [7] R. P. Curasma and H. P. Curasma, "Assessment and proposal of a network monitoring system based on free software," *Proc. 2020 IEEE Eng. Int. Res. Conf. EIRCON 2020*, Oct. 2020, doi: 10.1109/EIRCON51178.2020.9254047.
- [8] I. Saputra and D. Wiharta, "Implementasi Sistem Pemantauan Jaringan Menggunakan Librenms Pada Jaringan Kampus Universitas Udayana," *J. SPEKTRUM*, vol. 7, no. 2, pp. 81–82, 2020, Accessed: Jul. 08, 2022. [Online]. Available: [http://download.garuda.kemdikbud.go.id/article.php?article=1692923&val=955&title=IMPLEMENTASI SISTEM PEMANTAUAN JARINGAN MENGGUNAKAN LIBRENMS PADA JARINGAN KAMPUS UNIVERSITAS UDAYANA](http://download.garuda.kemdikbud.go.id/article.php?article=1692923&val=955&title=IMPLEMENTASI%20SISTEM%20PEMANTAUAN%20JARINGAN%20MENGUNAKAN%20LIBRENMS%20PADA%20JARINGAN%20KAMPUS%20UNIVERSITAS%20UDAYANA)
- [9] A. T. P. Afandi, "Implementasi Network Monitoring System Menggunakan Librenms Berbasis Docker Container".
- [10] T. Agus, "Monitoring Jaringan Internet Menggunakan Notifikasi Bot API Telegram," *SATIN - Sains dan Teknol. Inf.*, vol. 7, no. 1, pp. 144–153, Jun. 2021, doi: 10.33372/STN.V7I1.713.
- [11] L. V. Dasanty and D. A. Dermawan, "STUDI LITERATUR MONITORING MANAJEMEN JARINGAN INTERNET DENGAN KONSEP SNMP TERHADAP AKSES SISWA," *IT-Edu J. Inf. Technol. Educ.*, vol. 5, no. 01, pp. 38–48, 2020.
- [12] K. B. A. Nurcahyo and A. Prihanto, "Analisis Quality of Service (QoS) pada Jaringan VLAN (Virtual Local Area Network)".
- [13] E. Budiman, R. Wardhana, H. J. Setyadi, G. M. Putra, and E. Maria, "Network Traffic WLAN Monitoring based SNMP using MRTG with Erlang Theory," *3rd 2021 East Indones. Conf. Comput. Inf. Technol. EICONCIT 2021*, pp. 391–394, Apr. 2021, doi: 10.1109/EICONCIT50028.2021.9431898.
- [14] N. Chechina, H. Li, A. Ghaffari, S. Thompson, and P. Trinder, "Improving the network scalability of Erlang," *J. Parallel Distrib. Comput.*, vol. 90–91, pp. 22–34, Apr. 2016, doi: 10.1016/J.JPDC.2016.01.002.
- [15] A. Hidayat and A. Rizki, "MONITOR JARINGAN KOMPUTER BERBASIS WEB MENGGUNAKAN CACTI," *J. Tek. Inform.*, vol. 8, no. 1, Jul. 2020, doi: 10.51530/JUTEKIN.V8I1.439.