

# Implementation of the Apache Web Server Monitoring System Using LibreNMS at Diskominfo Sidoarjo Regency

## [Implementasi Sistem Monitoring Apache Web Server Menggunakan LibreNMS di Diskominfo Kabupaten Sidoarjo]

Hendrik Kusnanto<sup>1)</sup>, Hindarto Hindarto <sup>\*,2)</sup>

<sup>1)</sup> Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

<sup>2)</sup> Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

\*Email Penulis Korespondensi: 201080200199@umsida.ac.id<sup>1</sup>, hindarto@umsida.ac.id<sup>2</sup>

**Abstract.** The government is competing to make an innovation to facilitate public services. One of them is to create a web-based information system. Diskominfo is a Regional Apparatus Organization (OPD) whose job is to provide a data center within the Sidoarjo Regency Government. In this data center there are several web-based information systems, one of which is the SIPRAJA application. The problem occurs when this application is accessed by a large number of users. Identification of problems is the role of the administrator, usually the administrator checks server resources, network traffic and application code which is done manually one by one. This application uses Apache as its Web Server. To support the smooth running of an application, regular monitoring and systematic recording should be carried out. There are various monitoring applications that we can use for free, one of which is LibreNMS. Besides having a pretty attractive appearance, LibreNMS is also not limited to the number of devices being monitored. Therefore, in monitoring application servers, researchers use LibreNMS which has the ability to collect data via the SNMP protocol. Monitoring is done to make it easier for administrators to make decisions from identifying problems on the server. LibreNMS can be installed on a virtual server with the Linux Ubuntu 18.04 operating system which can produce monitoring in the form of graph data that can be easily analyzed. The method used in this research is to identify the problem, collect data, analyze the needs for system design, then implement it. From the test results, the researchers succeeded in implementing the Apache web server monitoring system in the form of displaying graphical data needed by administrators as reference material in decision making. The data displayed includes: Apache Scoreboard, Hits/sec, CPU Utilization, Traffic, Processor Usage, RAM Usage, Disk Usage.

**Keywords** - Apache; Data Center; LibreNMS; Monitoring; Protocol SNMP; Web Server

**Abstrak.** Pemerintah berlomba-lomba membuat sebuah inovasi untuk mempermudah pelayanan masyarakat. Salah satu-nya ialah membuat sistem informasi berbasis web. Diskominfo merupakan Organisasi Perangkat Daerah (OPD) yang bertugas menyediakan data center di lingkup Pemerintah Kabupaten Sidoarjo. Pada data center ini terdapat beberapa sistem informasi berbasis web salah satunya aplikasi SIPRAJA. Masalah terjadi Ketika aplikasi ini diakses dengan jumlah user yang banyak. Identifikasi dalam masalah merupakan peran dari administrator, biasanya administrator melakukan pengecekan resource server, traffic jaringan dan kode aplikasi yang dilakukan secara manual satu persatu. Aplikasi ini menggunakan Apache sebagai Web Server-nya. Untuk mendukung kelancaran dari sebuah aplikasi hendaknya dilakukan monitoring secara berkala dan tercatat secara sistematis. Ada beragam aplikasi monitoring yang dapat kita gunakan secara gratis salah satunya ialah LibreNMS. Selain memiliki tampilan yang cukup menarik LibreNMS juga tidak diberi batasan jumlah perangkat yang dimonitoring. Oleh karena itu dalam pemantauan server aplikasi, peneliti menggunakan LibreNMS yang memiliki kemampuan collect data melalui protocol SNMP. Pemantauan dilakukan agar memudahkan administrator dalam mengambil keputusan dari identifikasi masalah dalam server. LibreNMS dapat diinstall pada virtual server dengan operating system Linux Ubuntu 18.04 yang dapat menghasilkan pemantauan berupa grafik data yang bisa dengan mudah untuk dianalisa. Metode yang digunakan dalam penelitian ini adalah mengidentifikasi masalah, melakukan pengumpulan data, menganalisa kebutuhan untuk dilakukan perancangan sistem, kemudian di implemetasikan. Dari hasil pengujian peneliti berhasil mengimplementasikan sistem monitoring Apache web server berupa tampilan data grafik yang dibutuhkan administrator sebagai bahan rujukan dalam pengambilan keputusan. Data yang ditampilkan meliputi : Apache Scoreboard, Hit/sec, CPU Utilisation, Traffic, Processor Usage, RAM Usage, Disk Usage. .

**Kata Kunci** - Apache; Data Center; LibreNMS; Monitoring; Protocol SNMP; Web Server

## I. PENDAHULUAN

Demi mewujudkan kesejahteraan masyarakat pemerintah berlomba-lomba untuk melakukan inovasi yang bisa mempermudah dalam pelayanan masyarakat [1]. Hal yang paling banyak dilakukan oleh pemerintah di era ini, yaitu melakukan implementasi sistem informasi berbasis web. Salah satunya dilakukan oleh pemerintah kabupaten sidoarjo.

Dinas Komunikasi dan Informatika Kabupaten Sidoarjo merupakan Organisasi perangkat daerah (OPD) yang bertugas untuk melakukan pengelolaan jaringan komputer dan data center di lingkup pemerintah kabupaten sidoarjo [2]. Fasilitas yang disediakan oleh dinas ini berupa layanan akses informasi yang begitu luas meliputi seluruh OPD yang ada di kabupaten sidoarjo. Berupa jaringan komputer yang terhubung antar OPD serta penyediaan *server* untuk implementasi beberapa sistem informasi.

Pada Bagian Pemerintah Sekretariat Daerah yang di fasilitasi Dinas Komunikasi dan Informatika membuat sistem informasi kependudukan yang di beri nama SIPRAJA [3]. SIPRAJA menyediakan beberapa layanan untuk masyarakat salah satunya yaitu pembuatan pengajuan surat keterangan usaha. Surat keterangan ini nantinya digunakan masyarakat untuk melakukan pendaftaran usaha mikro di dinas perdagangan. Pada saat di bukanya fitur ini user (masyarakat) secara bersamaan mengakses aplikasi tersebut. Masyarakat sangat antusias dalam mengakses aplikasi ini, banyaknya user yang mengakses mengakibatkan aplikasi menjadi lambat. Aplikasi ini berbasis *web* dan menggunakan *Apache* sebagai *web server*-nya.

Disinilah peran administrator melakukan identifikasi terhadap sistem informasi yang telah dibuat. Pembuatan sistem informasi tentunya tidak hanya bisa berjalan sesuai dengan fungsinya tetapi juga harus memiliki kestabilan dan kelancaran dalam penerapannya [4]. Beberapa hal telah dilakukan oleh administrator di dinas komunikasi dan informatika kabupaten sidoarjo diantaranya melakukan pengecekan *bandwidth*, *resource server*, serta beberapa pengecekan *source code* dari aplikasi tersebut. Dalam identifikasi administrator membutuhkan data yang akurat dan terdokumentasi dengan baik guna untuk pengambilan sebuah keputusan.

Untuk memudahkan hal tersebut, Monitoring *server* menggunakan *LibreNMS* dapat mempermudah dalam menemukan masalah dan juga dapat mengetahui kapasitas ideal yang di butuhkan dalam sistem jaringan komputer dan *server* yang ada di lingkup Pemerintah Kabupaten Sidoarjo. *LibreNMS* memiliki kemampuan Collect Data melalui Protocol SNMP yang di aktifkan di perangkat yang akan di monitoring [5]. Salah satu yang dibutuhkan adalah monitoring *Apache Web Server*, dengan adanya fitur ini kita bisa mengetahui jumlah user yang mengakses *server* dan kita juga bisa mengetahui kapasitas maksimum dari *server* tersebut. Sehingga kita bisa menentukan resource yang dibutuhkan oleh *server* tersebut.

Perbedaan dengan penilitan sebelumnya Implementasi Sistem Pemantauan Jaringan Menggunakan *LibreNMS* pada Jaringan Kampus Universitas Udayana (Saputra, 2020) menggunakan *LibreNMS* untuk monitoring jaringan. Pada penilitan ini penulis menggunakan *LibreNMS* untuk monitoring *web server* khususnya *Apache*. Hal tersebut mendasari penulis untuk mengambil judul penelitian “ Implementasi Sistem Monitoring *Apache Web Server* Menggunakan *LibreNMS* di Diskominfo Kabupaten Sidoarjo”..

*Administrator* adalah orang yang bertanggung jawab dalam mengelola jaringan *computer* dan *server*, Salah satu tugas dari seorang administrator adalah monitoring *server* karena kesalahan pada *server* dan service yang berjalan di *server* tidak dapat diketahui kapan terjadi[6]. Untuk mempermudah administrator dalam melakukan pemantauan perangkat jaringan dan *server* maka dibutuhkan sistem yang dapat melakukan monitoring dengan baik dan tercatat secara sistematis[7].

*Monitoring* adalah suatu kegiatan pemantauan yang mencakup pengumpulan data , peninjauan ulang, pelaporan data, dan Tindakan atas informasi suatu proses atau objek yang sedang diimplementasikan [8]. Monitoring sangat diperlukan untuk menentukan tindakan menuju kearah yang lebih baik.

*Monitoring* dilakukan guna untuk mengetahui kondisi perangkat baik itu hardware ataupun software dari suatu arsitektur jaringan dan *server* [9]. Upaya monitoring kondisi perangkat sangat membantu dalam mengetahui performa dan bisa digunakan untuk mengidentifikasi suatu masalah dalam arsitektur jaringan dan *server*.

*Opensource* adalah istilah yang digunakan untuk software yang membebaskan source codenya untuk dilihat oleh orang lain dan membiarkan orang lain mengetahui cara kerja software tersebut dan sekaligus memperbaiki kelemahan-kelemahan yang ada pada software tersebut[10].

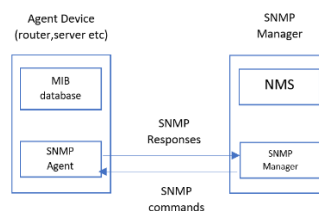
*Network Monitoring System* (NMS) merupakan tool untuk melakukan kegiatan monitoring kinerja dari perangkat jaringan dan *server*. NMS menggunakan *protocol SNMP* dalam pengumpulan datanya melalui kontrol jarak jauh [11]. Data hasil monitoring tersebut bisa digunakan untuk kepentingan identifikasi masalah ataupun bahan pertimbangan untuk perbaikan yang berkesinambungan. Selain itu monitoring juga bisa digunakan untuk meramalkan perubahan untuk perkembangan yang akan datang dan juga untuk mendeteksi perubahan yang tidak terduga yang mungkin terjadi, misal penyebab error pada *server* yang sedang berjalan, load resource *server* yang tinggi, traffic jaringan dll. Seorang administrator hanya mampu beraksi terhadap problem jika tanpa memiliki kemampuan memantau jaringan.

*Simple Network Management Protocol* (SNMP) adalah standard protokol yang digunakan untuk monitoring perangkat jaringan dan *server* melalui jarak jauh [12]. Informasi dari suatu perangkat baik itu hardware ataupun software di dapat dengan menggunakan protocol ini.

SNMP terdiri dari 3 bagian :

- *MIB*
- *SNMP Agent*
- *SNMP Manager*

MIB memiliki tugas untuk struktur database variable dari perangkat jaringan yang akan dimonitoring. Informasi nilai setiap variabel dapat diketahui atau di set dengan mudah karena struktur ini bersifat hierarki. *SNMP Agent* merupakan software yang dijalankan di setiap perangkat jaringan yang akan dimonitor. Memiliki tugas untuk mengumpulkan data atau informasi yang telah ditentukan di dalam MIB [11]. Sedangkan *SNMP Manager* bertugas untuk mengelola data yang di kirim melalui *SNMP agent* dan dikelola di *server NMS*. Gambar 1 merupakan struktur dari *SNMP* tersebut.



**Gambar 1.** Struktur SNMP

*LibreNMS* merupakan sistem aplikasi monitoring arsitektur jaringan berbasis web menggunakan *PHP*, *MariaDB* dan *protocol SNMP* pada penerapannya. *LibreNMS* merupakan aplikasi bersifat *open-source* sehingga tidak diperlukan *licensi*. Selain itu aplikasi ini juga bisa dilakukan modifikasi sedemikian rupa untuk menjadi lebih baik lagi. *LibreNMS* memiliki banyak sekali fitur monitoring salah satu diantaranya yang akan digunakan pada penelitian ini adalah fitur monitoring *Apache web server*.

Fitur-fitur yang tersedia di *LibreNMS* :

- Mendukung auto discover lokasi dari perangkat yang telah ditentukan longitude dan latitude nya.
- Fitur dashboard yang bisa di sesuaikan dengan kebutuhan.
- Mendukung API yang bisa di gunakan untuk mengambil data pada aplikasi mobile monitoring.
- Memungkinkan penggunaan user lebih dari satu.
- Perangkat yang dimonitoring ditampilkan dalam MAP.
- Mendukung pemantauan resource perangkat (CPU, RAM, HDD dan lain-lain).
- Mendukung semua perangkat yang bisa di monitoring melalui protokol SNMP.
- Memiliki UI dan UX yang menarik dan mudah dalam pengoperasiannya.
- Memiliki fitur monitoring *server* seperti *Apache*, *Ceph*, *Squid*, *Nginx*, *BIND* dll

*Apache* merupakan software web *server* yang open source, dikembangkan berbasis NCSA HTTPD 1.3 [13] yang kemudian di program ulang menjadi sebuah web *server* yang paling banyak digunakan saat ini. Sedangkan web *server* adalah software yang berfungsi untuk memberikan layanan berupa data dalam tampilan web. Pada penerapannya menggunakan protokol *HTTP* dan *HTTPS*. sebuah aplikasi atau website yang berbasis web memerlukan perna penting dari sebuah web *server*. Pada halaman atau page browser tidak akan bisa melakukan permintaan data apapun jika tanpa adanya web *server*.

*PuTTY* adalah aplikasi yang digunakan untuk melakukan kontrol jarak jauh dan biasanya menyediakan tipe koneksi berupa *Telnet*, *SSH*, *Raw*, maupun *Rlogin*. Yang bisa dilakukan di tempat yang berbeda. Dalam penelitian ini *PuTTY* berfungsi untuk melakukan instalasi melalui remote *server*.

## II. METODE

Metode penelitian yang digunakan dalam implementasi sistem monitoring ini yaitu identifikasi masalah, Analisa kebutuhan, pengumpulan data, skema instalasi sistem monitoring dan implementasi.

### A. Identifikasi Masalah

Penelitian dilaksanakan di Dinas Komunikasi dan Informatika (Diskominfo) Kabupaten Sidoarjo yang merupakan pusat layanan teknologi dan informasi di Kabupaten Sidoarjo. Secara khusus monitoring dilakukan pada *server* aplikasi SIPRAJA. Dimana pada aplikasi ini memiliki pengguna yang cukup banyak. Dan sering mengalami permasalahan (lambat) ketika di akses oleh user. Dalam pencarian masalah, Administrator masih menggunakan cara

manual dengan langsung melakukan pengecekan ke *server* aplikasi, yang dalam hal ini kurang efisien. Dikarenakan selain memakan waktu yang lama juga tidak adanya pencatatan yang sistematis. Dengan adanya sistem monitoring *web server* diharapkan bisa menjadi salah satu rujukan Administrator untuk mengidentifikasi masalah guna kelancaran dan kestabilan suatu sistem informasi berbasis web. Adapun data yang digunakan yaitu Traffic, Hits, CPU Utilisation, dan Scoreboard Statistics. Yang dihasilkan dari collect data aplikasi *LibreNMS* ini. Hasil dari *LibreNMS* ini akan dapat menjadi salah satu referensi untuk administrator dalam menentukan resource atau konfigurasi yang dilakukan pada *server* yang di monitoring. Bisa berupa penambahan Processor, RAM, ataupun tuneup di *Apache*.

## B. Analisa Kebutuhan

Dalam proses implementasi monitoring *web server* menggunakan *LibreNMS* ini ada beberapa komponen yang di implementasikan, berupa perangkat keras dan perangkat lunak.

Spesifikasi Perangkat keras yang digunakan pada penelitian ini adalah *Server Virtual* dengan spesifikasi sebagai berikut :

- Processor Intel Xeon Gold 6140 2,3 Ghz (2core)
- RAM 8 GB
- Harddisk 50GB

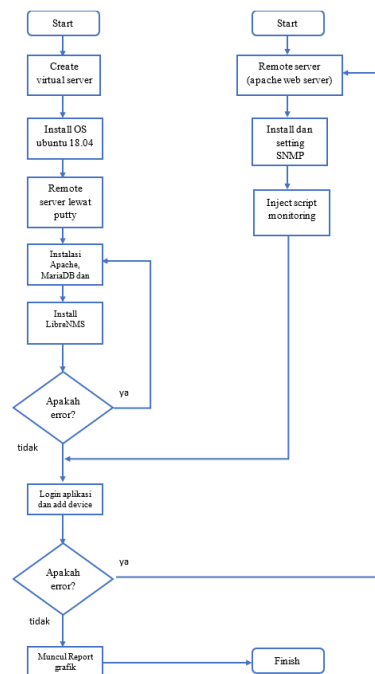
Sedangkan untuk perangkat lunak yang digunakan pada penelitian ini akan dijelaskan pada Tabel 1

**Tabel 1.** Perangkat Lunak Yang di Gunakan

Software	Keterangan
Hyper-V	Pembuat <i>server</i> virtual
Putty	Untuk remote <i>server</i> melalui SSH
Linux Ubuntu 18.04	Operating sistem untuk menjalankan aplikasi <i>LibreNMS</i>
LibreNMS	Aplikasi pemantau
MariaDB	Sistem manajemen database pendukung <i>LibreNMS</i>
Apache	Web <i>server</i> pendukung <i>LibreNMS</i>
PHP	Software pendukung untuk menjalankan aplikasi <i>LibreNMS</i>
Python	Software pendukung untuk menjalankan aplikasi <i>LibreNMS</i>
Chrome	Browser untuk menjalankan aplikasi

### C. Skema Instalasi Sistem Monitoring

Penelitian dimulai dari pembuatan *server* virtual melalui hyper-v yang sudah di install linux Ubuntu 18.04 dilengkapi dengan mengaktifkan *SSH*. Proses instalasi *LibreNMS* pada *server* tersebut menggunakan remote dengan aplikasi *PuTTY*. Adapun software yang dibutuhkan dalam implementasi *LibreNMS* ini yaitu *Apache*, *MariDB*, *Python* dan *PHP*. Setelah software pendukung di install maka selanjutnya melakukan instalasi *LibreNMS*. Tahap instalasi dan konfigurasi berhasil dilakukan kemudian melakukan penambahan IP atau subdomain dari *server* atau perangkat yang akan kita monitoring. Tentunya pada *server* atau perangkat tersebut sebelumnya sudah dilakukan aktivasi *SNMP*. Pada *Server LibreNMS* dilakukan pengecekan, pengujian resource dan *traffic server* atau perangkat yang di monitoring. Tahap penelitian diatas tertuang dalam skema pada Gambar 2 berikut ini.



Gambar 2. Skema Instalasi

### D. Pengumpulan Data

Sebagai bahan acuan dan penunjang dalam rancangan sistem monitoring *web server* ini diperlukan beberapa informasi yang oleh peneliti dibagi menjadi beberapa tahap yaitu :

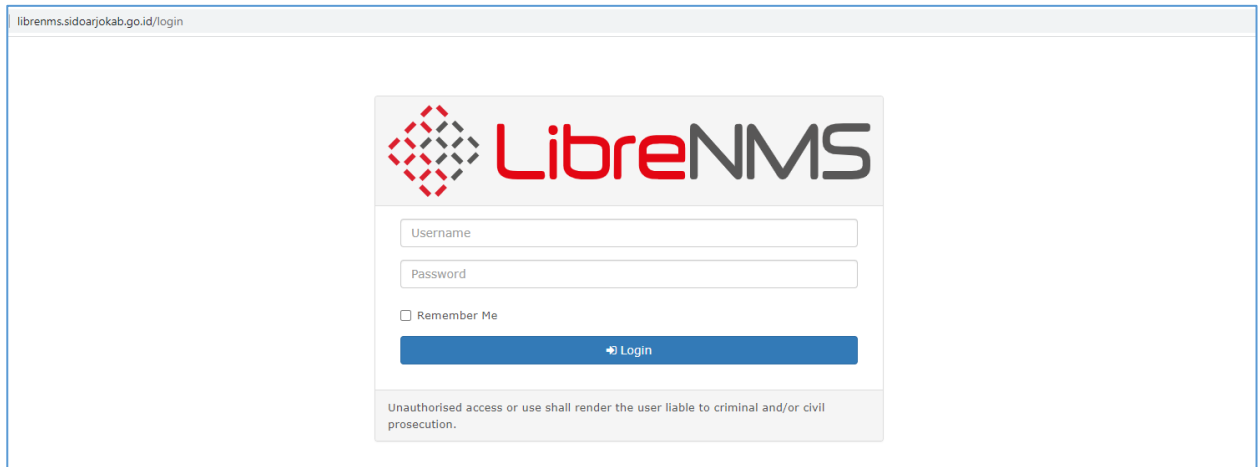
- Studi Pustaka, pengumpulan beberapa data, berbagai teori-teori yang terkait dengan penelitian sebagai pedoman untuk memperkuat referensi penelitian melalui jurnal penelitian terdahulu dan buku.
- Observasi, untuk mengetahui informasi rancangan sistem dilakukan pengamatan objek-objek secara langsung.
- Wawancara, mengumpulkan informasi tentang obyek dengan bertanya secara langsung pada pihak terkait yaitu Administrator.

### E. Implementasi LibreNMS

Tahapan-tahapan instalasi *LibreNMS* :

- Create virtual *server* di Hyper-V.
- Install OS ubuntu *server* 18.04 pada *server* virtual.
- Remote *server* ubuntu 18.04 tersebut menggunakan *PuTTY* melalui protokol *SSH*.
- Install perangkat lunak pendukung *LibreNMS* yang meliputi : *Apache*, *MariaDB*, *PHP* dan *Python*.
- Cloning aplikasi opensource *LibreNMS* melalui github.
- Konfigurasi database dan aplikasi *LibreNMS*.

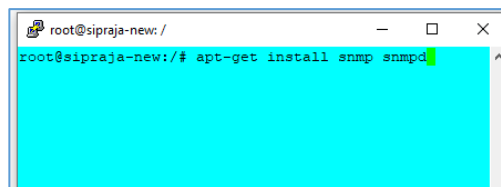
Setelah melakukan tahap instalasi. Aplikasi dapat di akses melalui *web browser* dengan cara mengakses *ip* atau *subdomain* pada *LibreNMS* tersebut. Pada Gambar 3 merupakan tampilan *user login* pada *LibreNMS*.



**Gambar 3.** Tampilan User Login LibreNMS

Sebelum dilakukan penambahan device yang akan di monitoring pada *LibreNMS* terlebih dahulu harus melakukan instalasi dan konfigurasi pada *Apache Web Server* (*server SIPRAJA*) yang akan dimonitoring. Tahapan yang dilakukan adalah :

- Instalasi SNMP dan SNMPd. Ketika perintah *apt-get install snmp snmpd* seperti pada gambar 4 berikut ini.



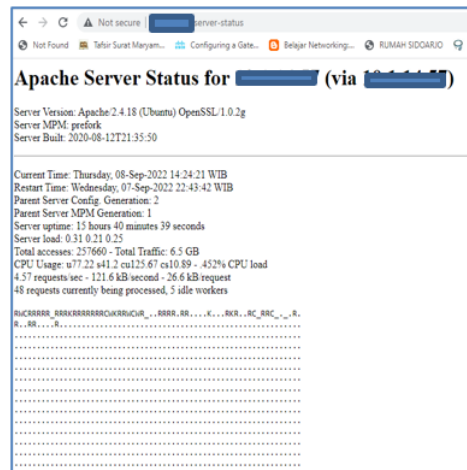
**Gambar 4.** Install SNMP SNMPD

- Lakukan konfigurasi SNMP tentukan *rocommunity* yang nantinya akan digunakan untuk add device pada aplikasi *LibreNMS* (*rocommunity* harus sama). Pada gambar 5 menambahkan *rocommunity* di file */etc/snmp/snmpd.conf* pada *server Apache*.



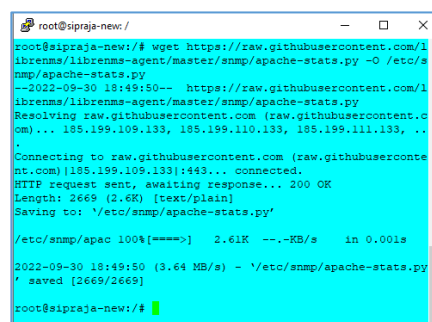
**Gambar 5.** Konfigurasi SNMP

- Aktifkan modul pada *Apache* *mod\_status*. Jika berhasil maka akan muncul tampilan *Apache Server Status* seperti pada gambar 6.



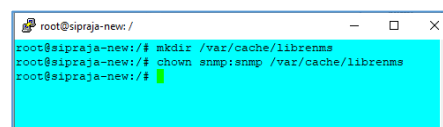
Gambar6. Apache Status Server

- Lakukan penambahan script untuk monitoring *Apache*, dengan perintah seperti Gambar 7.



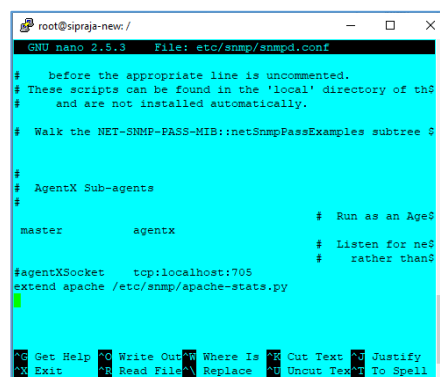
Gambar 7. Tampilan Cloning Script

- Tambahkan `chmod +x` pada file cloning tersebut.
- Buat folder di “ `var/cache/librenms/` “ dan rubah owner folder ke user `snmp` (user yang digunakan menjalankan service `snmp`) pada gambar 8.



Gambar 8. Buat Folder dan Rubah Owner

- Install `python3-urllib3`
- Tambahkan script `extend apache /etc/snmp/apache-stats.py` pada file `snmpd.conf` dan restart service `snmpd`. Sesuai pada gambar 9.

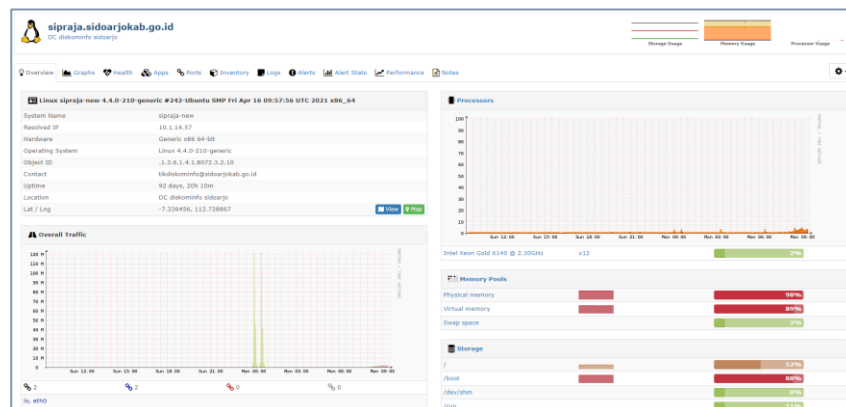


Gambar 9. Tambahan Script

Setelah melakukan instalasi dan konfigurasi pada *device /server* yang akan dimonitoring, maka konfigurasi selanjutnya yaitu menambahkan device pada aplikasi *LibreNMS*. Pada menu *add device hostname* merupakan alamat IP atau subdomain dari perangkat *server* yang akan dimonitoring, default untuk port SNMP yaitu 161 sedangkan *community* disesuaikan dengan konfigurasi SNMP yang akan dipantau. Gambar 10 merupakan tampilan dari menu *add device*.

**Gambar 10.** Tampilan Add Device di LibreNMS

Setelah dilakukan penambahan device, maka kondisi status *server* yang akan dipantau dapat dilihat di *LibreNMS* berupa grafik yang dapat dengan mudah untuk dianalisa, pemantauan tersebut sangat diperlukan oleh *administrator* dalam melakukan manajemen sebuah *server* [14]. Selain monitoring *Apache server* pada dasarnya *LibreNMS* juga menampilkan status *traffic*, *resource processor*, *memory*, *storage* bahkan *operating system* yang digunakan. Gambar 11 berikut merupakan tampilan kondisi *server* yang dipantau.



**Gambar 11.** Tampilan Kondisi Server

Setelah melakukan penambahan device (*server SIPRAJA*) aktifkan modul *Apache* pada menu Application yang ada di *LibreNMS* seperti tampilan pada gambar 12 berikut ini.

**Gambar 12.** Tampilan Aktif Modul *Apache*

### III. HASIL DAN PEMBAHASAN

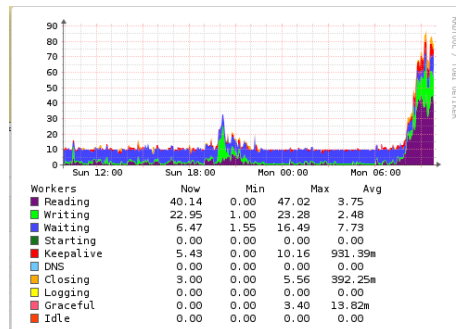
Pengujian sistem monitoring *LibreNMS* pada fungsionalitas monitoring web *server* menampilkan data-data berupa grafik. Dapat ditarik tabel hasil pengujian sebagai berikut :



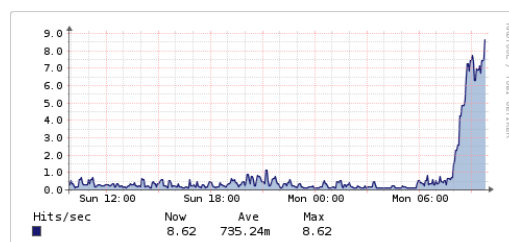
**Tabel 2.** Tabel Hasil Pengujian

Fungsionalitas	Yang di Uji	Hasil Pengujian
Fungsionalitas menampilkan data <i>hit/sec</i>	Interface <i>LibreNMS</i> dalam menampilkan data <i>hit/sec</i>	<i>LibreNMS</i> menampilkan Data <i>hit/sec</i> tampil berupa grafik
Fungsionalitas menampilkan data <i>CPU Utilisation</i>	Interface <i>LibreNMS</i> dalam menampilkan data <i>CPU Utilisation</i>	<i>LibreNMS</i> menampilkan Data <i>CPU Utilisation</i> tampil berupa grafik dengan satuan “%”
Fungsionalitas menampilkan data <i>Traffic</i>	Interface <i>LibreNMS</i> dalam menampilkan data <i>Traffic</i>	Data <i>traffic</i> tampil dalam satuan Mbps
Fungsionalitas menampilkan data <i>server-status</i>	Interface <i>LibreNMS</i> dalam menampilkan data <i>server-status</i>	<i>LibreNMS</i> menampilkan data <i>server-status</i> melalui grafik <i>Apache scoreboard</i>

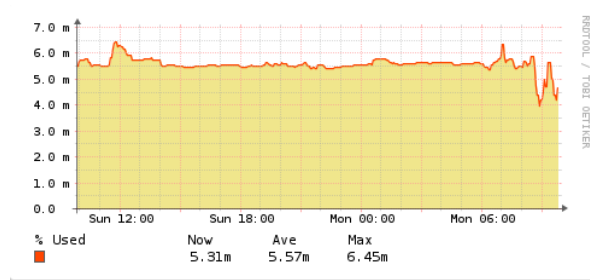
Pemantauan *Apache Web server* khususnya *Apache Scoreboard* yang dilakukan oleh aplikasi *LibreNMS* di Diskominfo kabupaten sidoarjo bisa di lihat pada Gambar 13. Dengan adanya data ini, Administrator bisa mengetahui bagaimana performa penggunaan *Server Apache*. Pada grafik ini terdapat data *Reading, Writing, Waiting, Starting, Keepalive, DNS, Closing, Logging, Graceful, Idle*.

**Gambar 13.** Tampilan Grafik *Apache Scoreboard*

Pada pemantauan trafik, *administrator* bisa mengidentifikasi pada saat kapan terjadi lonjakan data yang bisa mengakibatkan terjadi permasalahan pada aplikasi (*Server SIPRAJA*). Data ini juga bisa di gabungan dengan data *hit/sec* yang ada pada Gambar 14, untuk menjadikan pertimbangan oleh *administrator* dalam memecahkan masalah pada suatu aplikasi. Pada Gambar 14 ini adalah hasil pemantauan jumlah request akses aplikasi yang ditampilkan dalam satuan *hit/sec*. untuk sumbu vertical merupakan jumlah *hit/sec* sedangkan untuk horizontal adalah waktu akses user.

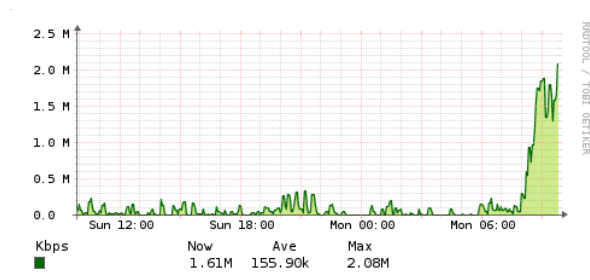
**Gambar 14.** Tampilan Grafik *hit/sec*

Pada Gambar 15 merupakan tampilan CPU Utilisation yang berfungsi untuk menampilkan seberapa besar penggunaan processor yang digunakan oleh *Apache* pada sistem yang ditampilkan dalam satuan persentase.



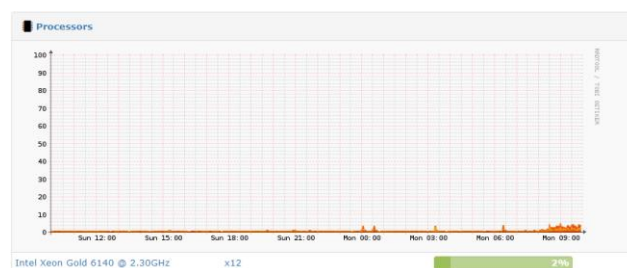
**Gambar 15.** Tampilan Grafik CPU Utilisation

*LibreNMS* juga memonitor traffic pengguna *Apache* yang ditampilkan dalam bentuk grafik dengan satuan Mbps. Dapat dilihat pada gambar 16.



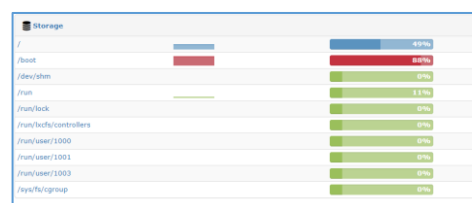
**Gambar 16.** Tampilan Grafik Traffic

Selain data grafik di atas pada *LibreNMS* juga bisa memantau resource dari *server* seperti penggunaan processor yang tampil pada gambar 17. Pada grafik ini bisa dilihat jumlah core pada *server* yang di monitoring dan penggunaannya dengan satuan “%”. Grafik ini bisa menjadi acuan *Administrator* untuk melakukan penambahan processor pada *server*.



**Gambar 17.** Grafik Penggunaan Processor

Selain penggunaan processor *LibreNMS* juga bisa menampilkan grafik penggunaan RAM terlihat pada gambar 18 berikut ini. Penggunaan RAM mencapai 99% yang artinya *Administrator* harus mempertimbangkan untuk penambahan RAM pada *server* tersebut.



**Gambar 19.** Penggunaan Harddisk

#### IV. SIMPULAN

Kesimpulan yang diperoleh dari implementasi system monitoring *Apache Web Server* menggunakan *LibreNMS* di Dinas Komunikasi dan Informatika Kabupaten Sidoarjo adalah sebagai berikut.

Dari hasil pembahasan system monitoring *LibreNMS* ini dapat menampilkan data berupa grafik yang memudahkan *Administrator Server* dalam menentukan suatu keputusan identifikasi suatu masalah.

Berdasarkan hasil pemantauan grafik performance *Apache Web Server* terlihat bahwa *Load Apache* sedang tinggi pada saat waktu tertentu. Kondisi ini bisa digunakan acuan untuk mengambil sebuah keputusan dalam identifikasi masalah. Bisa berupa penambahan resource server, setting *tuneup apache*, ataupun pengecekan *script* dari aplikasi yang berjalan di *server*.

#### UCAPAN TERIMA KASIH

Kami panjatkan puji syukur kehadiran ALLAH SWT yang telah memberikan berkah kepada penulis sehingga telah menyelesaikan penelitian ini. Terima kasih untuk Bapak dan Ibu Dosen Informatika Umsida yang telah memberikan ilmu selama perkuliahan. Ucapan terima kasih kepada kedua orang tua, keluarga, dan seluruh pihak terkait yang telah berkontribusi dalam proses penelitian ini, sehingga jurnal ilmiah ini telah terselesaikan dengan baik.

#### REFERENSI

- [1] N. Putu, D. Rosalina, and H. Narsa, "Inovasi Pelayanan: Telaah Literatur Perbandingan Sektor Privat dan Sektor Publik," 2018.
- [2] yohanes siswoyo, "renstra diskominfo," Jul. 2018.
- [3] Pemerintah Kabupaten Sidoarjo, *Surat Edaran Bupati Sidoarjo tentang Pelayanan dan Perijinan Desa dan Kecamatan di Kabupaten Sidoarjo*. indonesia, 2020, pp. 1–3. [Online]. Available: [www.sidoarjokab.go.id](http://www.sidoarjokab.go.id)
- [4] N. : Khadijah and Z. Nurbana, "Aplikasi Pada Ponsel Android Untuk Administrasi User Jaringan Melalui Short Message Service," 2013.
- [5] I. Wayan Krisna Saputra, D. Made Wiharta, and N. Putra Sastra, "Implementasi Sistem Pemantauan Jaringan Menggunakan Librenms Pada Jaringan Kampus Universitas UDAYANA," *Jurnal SPEKTRUM*, vol. 7, no. 2, pp. 81–89, 2020.
- [6] M. A. Husna and P. Rosyani, "Implementasi Sistem Monitoring Jaringan dan Server Menggunakan Zabbix yang Terintegrasi dengan Grafana dan Telegram," *Jurnal Riset Komputer*, vol. 8, no. 6, pp. 2407–389, 2021, doi: 10.30865/jurikom.v8i6.3631.
- [7] R. Rinaldo, "IMPLEMENTASI SISTEM MONITORING JARINGAN MENGGUNAKAN MIKROTIK ROUTER OS DI UNIVERSITAS ISLAM BATIK SURAKARTA," *Jurnal Teknik Elektro*, vol. 16, no. 02, pp. 56–63, 2016.
- [8] M. S. Firmansyah, "Monitoring Web Server Berbasis Jaringan Cisco Catalyst Series 2960 Menggunakan Librenms Di PT. Telekomunikasi Indonesia DIVRE V Jatim," 2019.
- [9] Suherman and L. Sahidin, "Monitoring Perangkat Teknologi Informasi Berbasis IP dengan Integrasi SMS Gateway," *Jurnal PROSISKO*, vol. 2, no. 2, pp. 41–50, 2015.
- [10] J. Yanto and M. Ruswanda, "Implementasi Sistem Monitoring Server Menggunakan Nagios," 2017. [Online]. Available: <http://www.i-tech.ac.id>
- [11] Y. Ardian, "Simple Network Monitoring Protocol (SNMP) Untuk Memonitor Trafik User Studi Kasus: Universitas Kanjuruhan Malang," *STIKI Informatika Jurnal*, vol. 05, no. 01, pp. 20–24, 2015.
- [12] F. Maulana, "Implementasi Simple Network Management Protocol (Snmp) Pada Aplikasi Monitoring Jaringan Berbasis Website(Studi Kasus Universitas Muhammadiyah Bengkulu)," *Jurnal Informatika*, vol. 16, no. 2, 2016.
- [13] A. Y. Chandra, "Analisis Performansi Antara Apache & Nginx Web Server Dalam Menangani Client Request," *Jurnal Sistem dan Informatika (JSI)*, vol. 14, no. 1, pp. 48–56, Nov. 2019, doi: 10.30864/jsi.v14i1.248.
- [14] S. Taftazanie, A. Budi Prasetyo, and E. Didik Widiyanto, "Aplikasi Pemantau Perangkat Jaringan Berbasis Web Menggunakan Protokol SNMP dan Notifikasi SMS," *Jurnal Teknologi dan Sistem Komputer*, vol. 5, no. 2, pp. 62–68, 2017, doi: 10.14710/jtsiskom.5.2.2017.62-68.

#### **Conflict of Interest Statement:**

*The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.*