

# IMPLEMENTASI JARINGAN HOTSPOT BERBAYAR BERBASIS VOUCHER MENGGUNAKAN PLATFORM GOOGLE CLOUD

## *Implementation of A Paid Hotspot Network Based on Vouchers Using the Google Cloud Platform.*

Javier Aditama Falaq<sup>1</sup>, Rohmat Tulloh<sup>2</sup>, Muhammad Iqbal<sup>3</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

javieraditama@student.telkomuniversity.ac.id<sup>1</sup>, rohmatth@telkomuniversity.ac.id<sup>2</sup>,  
muhiqbal@telkomuniversity.ac.id<sup>3</sup>

**Abstract** - Hotspot is an area where people can access the internet network. If the hotspot application is not only in one place, it is difficult to manage and monitor the network because the network admin has to be at the hotspot to manage and monitor the network. In this final project, a voucher-based hotspot network has been esigned using the Google Cloud platform, which uses Usermanager, Zabbix and Grafana as a network management and monitoring system, and the PCQ method as a bandwidth management for hotspot users. Throughput testing using the PCQ method has a slightly smaller amount of 6.1142 MB / s and the throughput without the PCQ method is 8.2918 Mbps. The packet loss using the PCQ method has 1.8677% of examiners, slightly larger than without using the PCQ method, namely 1.8198%. The delay test using the PCQ method obtained 1.077625 ms which was slightly greater than that without the PCQ method, namely 0.838202 ms. In testing the jitter using the PCQ method resulted in 1.077639 ms, slightly greater than that without the PCQ method, namely 0.838229 ms.

**Keywords** – Hotspot, Wi-Fi, Zabbix, Grafana, Google Cloud Platform

**Abstrak** - Hotspot merupakan area dimana orang dapat mengakses jaringan internet. Apabila dalam pengaplikasian hotspot tidak hanya satu tempat, sulit dalam melakukan manajemen dan memonitor jaringan dikarenakan admin jaringan harus berada di lokasi hotspot untuk manajemen dan memonitoring jaringan. Pada proyek akhir ini, telah dirancang jaringan hotspot berbasis voucher dengan memanfaatkan platform Google Cloud, di mana menggunakan Usemanager, Zabbix dan Grafana sebagai sistem manajemen dan monitor jaringan, metode PCQ sebagai manajemen bandwidth pengguna hotspot.

**Pengujian throughput dengan menggunakan metode PCQ memiliki sedikit lebih kecil yaitu sebesar 6.1142 MB/s dan pada throughput tanpa metode PCQ sebesar 8.2918 Mbps. Pada pengujian packet loss dengan menggunakan metode PCQ memiliki sebesar 1.8677 % sedikit lebih besar dibandingkan tanpa menggunakan metode PCQ yaitu 1.8198 %. Pengujian delay dengan menggunakan metode PCQ memperoleh 1.077625 ms sedikit lebih besar dibanding tanpa metode PCQ yaitu 0.838202 ms. Pada pengujian jitter dengan menggunakan metode PCQ memperoleh 1.077639 ms sedikit lebih besar dibanding dibanding tanpa metode PCQ yaitu 0.838229 ms.**

**Kata Kunci** – Hotspot, Wi-Fi, Zabbix, Grafana, Google Cloud Platform

### I. PENDAHULUAN

Zaman sekarang teknologi sudah berkembang dengan sangat pesat dan dampaknya membuat kebutuhan manusia untuk penggunaan internet semakin tinggi. Saat ini penggunaan jaringan komputer menggunakan wifi sudah banyak dijumpai di berbagai tempat, seperti perkantoran, sekolah, kafe dan sebagainya. Dengan besarnya pengguna dalam mengakses layanan internet, maka dari itu dapat dijadikan sebagai ladang usaha baru dengan membuka fasilitas *hotspot*. Keuntungan dalam bidang ini pun relatif tinggi dikarenakan masih jarang ditemukan usaha bisnis *hotspot* dan harga yang ditawarkan masih relatif terjangkau.

Dengan semakin banyaknya layanan akses internet publik berbasis *hotspot* yang menggunakan sistem paket voucher, akan muncul permasalahan baru dari sisi keefektifitasannya. Apabila layanan akses internet *hotspot* terdapat lebih dari satu tempat, maka dalam melakukan konfigurasi dan manajemen terhadap *user hotspot* dan perangkat jaringan harus dilakukan pada tiap lokasi *hotspot* berada.

<sup>\*)</sup>Penulis korespondensi (Javier Aditama Falaq)  
Email: [javieraditama@student.telkomuniversity.ac.id](mailto:javieraditama@student.telkomuniversity.ac.id)

Dengan menggunakan *Cloud Computing*, dapat membuat sebuah perangkat virtual yang kemudian akan digunakan untuk membangun sebuah sistem manajemen dan *monitoring* jaringan. Salah satu contoh penggunaan *Cloud Computing* yaitu dengan menggunakan Google

*Cloud Platform*. Google *Cloud Platform* merupakan layanan *public cloud computing* dari Google yang terdiri dari beragam layanan *platform*. Dengan menggunakan layanan pada Google *Cloud Platform*, dapat membangun *virtual machine* secara instan. Pada Google *Cloud Platform* akan digunakan untuk membangun sistem manajemen *user hotspot* dan manajemen perangkat jaringan sehingga dalam melakukan manajemen dan *monitoring* user hotspot dan perangkat jaringan dapat dilakukan dimana saja tanpa perlu dilakukan di lokasi hotspot berada.

Pada penelitian [1] pada pengaplikasiannya hanya dapat *memonitoring* klien yang terhubung dengan jaringan dan tidak terdapat sistem *monitoring* perangkat jaringan. Pada penelitian [2] pada pengaplikasiannya hanya menyediakan satu lokasi jaringan hotspot sebagai layanan internet. Pada penelitian [3] pada pengaplikasiannya masih menggunakan jaringan konvensional, tidak menggunakan teknologi *cloud computing*. Pada penelitian [4] pada pengaplikasiannya tidak menggunakan sistem *Radius Server*. Pada penelitian [5] pada pengaplikasiannya masih menggunakan jaringan konvensional, tidak menggunakan teknologi *cloud computing*. Pada penelitian [6] pada pengaplikasiannya tiap akun tidak menerapkan sistem kuota.

## II. PERENCANAAN SISTEM

### 1. Hotspot

*Hotspot* adalah satu standar *Wireless Networking* tanpa kabel, hanya dengan komponen yang sesuai dapat terkoneksi ke jaringan (*Wireless Local Area Network*) [7]. Pada dasarnya *hotspot* merupakan sistem yang memberikan fitur autentikasi pada *user* yang akan terhubung ke sebuah jaringan. Untuk bisa akses ke jaringan dibutuhkan *username* dan *password* pada *login page*.

### 2. Cloud Computing

*Cloud Computing* (komputasi awan) adalah teknologi yang memanfaatkan layanan internet menggunakan pusat *server* yang bersifat virtual dengan tujuan pemeliharaan data dan aplikasi. Komputasi awan merupakan suatu konsep umum dari perkembangan *web 2.0* yang disinergikan dengan komputer-komputer pengguna (*device*) dengan teknologi terbaru yang mampu mengakses internet kapanpun dan di manapun, sehingga mempermudah pengguna tetap berinteraksi dengan data dan dokumen, dan di saat yang sama berbagi data dan dokumen tersebut dengan rekan kerja [8].

### 3. Radius

*Radius (Remote Authentication Dial-In User Service)* merupakan sebuah *protocol* yang digunakan untuk melakukan autentikasi, otorisasi, dan pendaftaran akun pengguna secara terpusat untuk mengakses jaringan. *Server Radius* menyediakan mekanisme autentikasi dan otorisasi koneksi yang dilakukan *user*. Pada klien ingin menghubungkan diri ke sebuah jaringan, maka *Server Radius* akan meminta *username* dan *password* yang kemudian kelak data tersebut akan dicocokkan dengan data yang terdapat pada *database Server Radius* untuk ditentukan *user* tersebut diijinkan menggunakan layanan tersebut [1].

### 4. User Manager

*User manager* merupakan fitur *AAA Server* yang dimiliki mikrotik. *User Manager* memiliki *database* yang bisa digunakan untuk autentikasi *user* yang *login* ke dalam suatu jaringan memberikan kebijakan terhadap *user* seperti memberikan limitasi kecepatan *bandwidth* serta batasan kuota yang diberikan. *User Manager* ini memudahkan untuk membuat layanan *internet public* secara luas [1].

### 5. Mikrotik Router OS

MikroTik *RouterOS* merupakan sistem operasi yang diperuntukkan sebagai *network router*. MikroTik *routerOS* sendiri adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan komputer biasa menjadi *router network* yang handal, mencakup berbagai fitur yang dibuat untuk *ip network* dan 8 jaringan *wireless*. Fitur-fitur tersebut diantaranya: *Firewall & Nat*, *Routing*, *Hotspot*, *Point to Point Tunneling Protocol*, *DNS server*, *DHCP server*, *Hotspot*, dan masih banyak lagi fitur lainnya. MikroTik *routerOS* merupakan sistem operasi *Linux base* yang diperuntukkan sebagai *network router*. Didesain untuk memberikan kemudahan bagi penggunaannya [9].

### 6. Google Cloud Platform

Menurut (Challita et al. 2018) “Di antara *Cloud API*, *Google Cloud Platform (GCP)* adalah salah satu yang paling penting dan berkembang di *cloud* pasar. Memberikan *developers* beberapa produk berbagai program dari situs *website* sederhana hingga aplikasi terdistribusi yang kompleks di seluruh dunia. *Google Cloud Platform* atau *GCP* adalah layanan *public cloud* komputasi dari *Google LLC* yang terdiri dari berbagai layanan. *Platform* ini menyediakan berbagai macam layanan *hosting* mulai dari komputasi, penyimpanan dan pengembangan aplikasi yang berjalan di perangkat keras *Google*. Layanan *Google Cloud Platform* dapat diakses dengan menggunakan *software* pengembang, *administrator cloud*, dan *TI profesional* lainnya yang menggunakan internet publik atau melalui koneksi jaringan khusus [10].

### 7. Zabbix

Menurut Santosa (2010), *Zabbix* merupakan aplikasi pemantauan ketersediaan dan performa jaringan komputer kode terbuka (*opensource*). Pada *zabbix* dapat menghasilkan grafis statistik, peta jaringan, *screen monitoring* dan notifikasi apabila terdapat perangkat yang mengalami masalah. *Zabbix* mudah di instalasi dan dikonfigurasi, pada distribusi *Linux Ubuntu*, *zabbix* secara *default* dimasukan ke *repository ubuntu*, pengguna hanya melakukan instalasi dengan *apt-get*. Terdapat 3 *software* utama dalam arsitektur *zabbix* yaitu *zabbix server*, *zabbix agent* dan *zabbix proxy*. *Zabbix server*

adalah proses utama dari *software* zabbix. *Zabbix agent*: *Zabbix agent* adalah UNIX *daemon* yang berjalan pada *host* yang sedang dipantau. Agen ini memberikan informasi, performa pada *host* dan mengirimkan informasi-informasi tersebut ke *Server zabbix*. *Zabbix proxy*: *Zabbix Proxy* adalah sebuah proses yang bertugas mengumpulkan hasil kinerja dan ketersediaan dari satu atau lebih peralatan yang dimonitor dan mengirimkan data tersebut kepada *Zabbix server* [11].

### 8. SSTP (Secure Socket Tunneling Protocol)

Merupakan bentuk virtual *private network* (VPN) yang menyediakan mekanisme untuk *transport traffic* PPP melalui channel SSL / TLS. SSL / TLS menyediakan keamanan level *transport* dengan *key negotiation*, enkripsi, dan *integrity checking*. Penggunaan SSL / TLS melalui *port* TCP 443 memungkinkan SSTP untuk melewati *h,32* hampir semua *firewall* dan *server proxy* kecuali untuk *proxy web* yang diautentikasi. *Server* SSTP harus diautentikasi selama fase SSL / TLS. Klien SSTP dapat secara opsional diautentikasi selama fase SSL / TLS dan harus diautentikasi dalam fase PPP. SSTP hanya ditujukan untuk akses klien jarak jauh. SSTP hanya mendukung otentikasi pengguna, itu tidak mendukung otentikasi perangkat atau otentikasi komputer [1].

### 9. Grafana

Grafana adalah sebuah *software opensource* yang membaca sebuah data *metrics* untuk dibuat menjadi sebuah grafik atau sebuah data tertulis. Grafana sering digunakan untuk melakukan analisis data dan *monitoring*. Grafana mendukung banyak *storage backends* yang berbeda untuk data *time series* (*Source Data*). Setiap *source data* memiliki *Query Editor* tertentu yang disesuaikan untuk fitur dan kemampuan tertentu. Grafana mendukung banyak basis data seperti Graphite, Prometheus, INfluxDB, ElasticSearch, MySQL, PostgreSQL. Grafana juga memungkinkan untuk menulis *plugin* untuk diintegrasikan dengan beberapa basis data yang berbeda. Salah satu keuntungan dari penggunaan grafana adalah Grafana tidak akan mengalirkan data ke *cloud vendor* untuk alasan keamanan dan lainnya. Dengan menggunakan grafana akan memudahkan untuk meminta, memvisualisasikan, mengatur peringatan dan memahami data dengan bantuan *metric* [12].

### 10. PCQ (Per Connection Queue)

PCQ adalah program untuk mengelola jaringan lalu lintas kualitas layanan (QoS). Tujuan utama dari metode ini adalah untuk melakukan *bandwidth sharing* otomatis dan merata ke multi klien. Kerja prinsip PCQ dengan menerapkan *simple queue* atau *queue trees* di mana hanya ada satu klien aktif yang menggunakan *bandwidth*, sementara klien lain berada dalam posisi *idle* maka klien aktif tersebut dapat menggunakan *bandwidth* maksimum yang tersedia, tetapi jika klien lain aktif, maka *bandwidth* yang maksimal dapat digunakan oleh kedua klien (*bandwidth* atau jumlah klien yang aktif) sehingga *bandwidth* dapat terdistribusi secara adil untuk semua klien.

Menurut Mujahidin dalam jurnal yang berjudul OS Mikrotik Sebagai Manajemen *Bandwidth* Dengan Menerapkan Metode *Per Connection Queue* (2011), menjelaskan tentang konfigurasi manajemen *bandwidth* dengan menggunakan metode *Per Connection Queue* (PCQ) dan menggunakan sistem antrean *queue tree*. Pada prinsipnya, penggunaan metode antrean untuk menyeimbangkan *bandwidth* yang digunakan pada beberapa klien [13].

### 11. SNMP (Simple Network Management Protocol)

*Simple Network Management Protocol* (SNMP) adalah sebuah protokol yang dirancang untuk memberikan kemampuan kepada pengguna untuk memonitor dan mengatur suatu jaringan komputer dari jarak jauh (secara *remote*) atau dalam satu pusat kontrol saja. Dengan menggunakan protokol ini bisa didapatkan informasi tentang status dan keadaan dari suatu jaringan. Protokol ini menggunakan *transport* UDP pada *port* 161. Pengolahan ini dijalankan dengan mengumpulkan data dan melakukan penetapan terhadap variabel-variabel dalam elemen jaringan yang dikelola [14]. Dalam aplikasinya, Elemen SNMP terdiri dari tiga bagian, yaitu *manager*, *agent*, dan MIB [15]. *Manager* merupakan *software* yang berjalan di sebuah *host* di jaringan, yang merupakan suatu proses atau lebih yang berkomunikasi dengan *agent* dalam jaringan. *Agent* merupakan perangkat lunak yang dijalankan disetiap elemen jaringan yang dikelola. *Agent* terdapat pada, *workstation*, *repeater*, *router*, *switch*, dan *personal computer*, bertugas untuk merespon dan memberikan informasi sesuai permintaan *manager* SNMP. *Manager Information Base* (MIB) merupakan struktur *database* variabel dari elemen jaringan yang dikelola [16]. Pendefinisian MIB dalam SNMP menggunakan diagram pohon, dan menempatkan setiap *Object Identifier* (OID) pada suatu lokasi unik pada pohon [17].

### 12. QOS (Quality of Service)

*Quality of Service* (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu layanan. QoS berfungsi untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu layanan [18].

Parameter yang digunakan:

#### a. *Throughput*

*Throughput* merupakan kecepatan *transfer* data efektif, yang diukur dalam bps (*bit per second*). *Throughput* adalah jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu [19].

Tabel 1. Parameter *Throughput*

Kategori <i>Throughput</i>	<i>Throughput</i> (%)
Sangat Bagus	100 - 75
Bagus	75 - 50
Sedang	50 - 25
Jelek	<25

(TIPHON,1999)

Persamaan perhitungan *Throughput*:

## b. Delay

Merupakan variasi dari *delay end-to-end*. *Level level* yang tinggi pada *jitter* dalam aplikasi-aplikasi berbasis UDP merupakan situasi yang tidak dapat diterima dimana aplikasi-aplikasinya merupakan aplikasi-aplikasi *real-time*. Pada kasus seperti itu, *jitter* akan menyebabkan sinyal terdistorsi, yang dapat diperbaiki hanya dengan meningkatkan *buffer* di antrian [19].

Tabel 2. Parameter *Delay*

Kategori	<i>Delay</i>	Indeks
Sangat Bagus	< 150 ms	4
Bagus	150 s/d 300 ms	3
Sedang	300 s/d 450 ms	2
Jelek	> 450 ms	1

(TIPHON, 1999)

Persamaan perhitungan *delay (latency)*:

## c. Jitter

Merupakan total waktu yang dilalui suatu paket dari pengirim ke penerima melalui jaringan. Delay dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama [19].

Tabel 3. Parameter *Jitter*

Kategori	<i>Jitter</i>	Indeks
Sangat Bagus	0 ms	4
Bagus	0 s/d 75 ms	3
Sedang	75 s/d 125 ms	2
Jelek	125 s/d 225 ms	1

(TIPHON, 1999)

Persamaan perhitungan *Jitter*:

## d. Packet Loss

Merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang, dapat terjadi karena *collision* dan *congestion* pada jaringan. Umumnya perangkat jaringan memiliki *buffer* untuk menampung data yang diterima. Jika terjadi kongesti yang cukup lama, *buffer* akan penuh, dan data baru tidak akan diterima [19].

Tabel 4. Packet Loss

Kategori	<i>Packet Loss</i>	Indeks
Sangat Bagus	0 %	4
Bagus	3 %	3
Sedang	15 %	2
Jelek	25 %	1

(TIPHON, 1999)

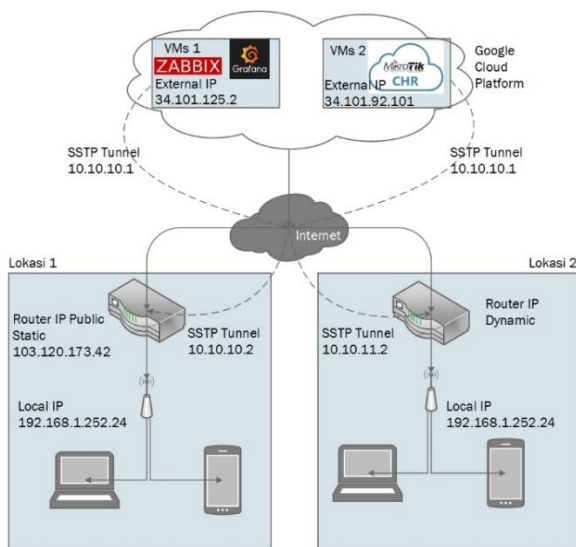
Persamaan perhitungan *Packet Loss*:

## 13. Wireshark

Wireshark merupakan salah satu aplikasi *open source* untuk mengetahui lalu lintas komunikasi data dalam jaringan dengan cara memantau melalui protokol dan *port-port* yang digunakan. Wireshark adalah salah satu dari sekian banyak *tool network analyzer* yang banyak digunakan oleh *network administrator* untuk menganalisa kinerja jaringannya. Wireshark banyak disukai karena interfacenya yang menggunakan *Graphical User Interface (GUI)* atau tampilan grafis. [20]. Wireshark mampu merekam semua paket data yang lewat serta menyeleksi dan menampilkan data secara akurat. Semua jenis paket data dalam berbagai format protokol apapun dapat dengan mudah direkam dan dianalisa.



IV. PERENCANAAN SISTEM



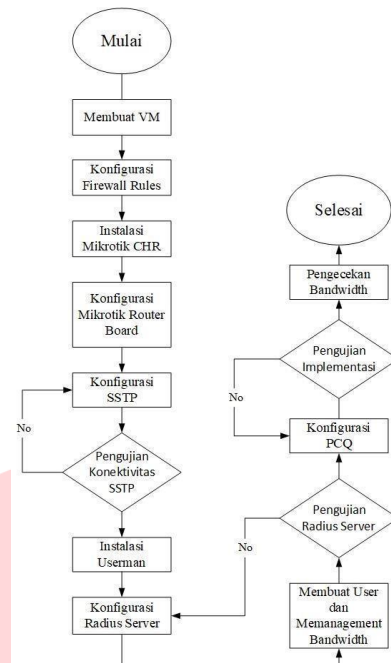
Gambar 1. Topologi Jaringan

Model sistem pada proyek akhir ini dapat dilihat pada Gambar 1 pada model ini terdapat 2 buah internet *service provider* yang digunakan yaitu *My Republic* pada *hotspot* lokasi 1 dan *Indihome* pada *hotspot* lokasi 2. Terdapat 2 buah mikrotik yang sudah dihubungkan ke mikrotik CHR dengan menggunakan protokol SFTP dan menggunakan sistem PCQ untuk mengatur sistem antrian pada pengguna *hotspot*. Terdapat 2 buah *virtual machine* yang berupa 1 buah *server* yang terdapat *Zabbix Server* dan *Grafana* sebagai sistem monitor dan 1 buah *router* yang terdapat *Userman* untuk membuat *voucher* dan mengatur *bandwidth* pada tiap pengguna *hotspot*. Dan akan dilakukan pengujian dengan menguji apakah sistem *voucher* dan *monitoring* dapat berjalan dan di aplikasikan.

Pada skenario ini tahapan *login user* dan sistem *monitoring* dapat diketahui sebagai berikut:

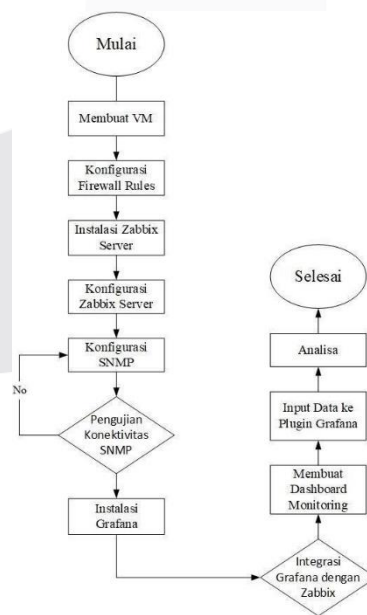
1. *Userman* akan digunakan untuk membuat *voucher* dan mengatur *bandwith* pada tiap *voucher* yang telah dibuat.
2. *Login user* pada tiap lokasi *hotspot* dengan menggunakan *voucher* yang telah dibuat pada *Userman*.
3. SFTP akan mengatur sistem antrian *download* dan *upload* pada *hotspot* yang telah tersedia.
4. *Zabbix Server* akan melakukan pengambilan data informasi pada tiap perangkat jaringan.
5. *Grafana* akan mengolah data yang dikirim dari *Zabbix Server* sehingga dapat membuat *dashboard* yang memudahkan administrator jaringan dalam melakukan *monitoring* setiap perangkat jaringan.

Pada perencanaan jaringan ini akan terdapat dua jenis tahapan perencanaan yang di lakukan. Diagram alir tahapan yang akan dilakukan, dapat dilihat pada Gambar 2.



Gambar 2. Flowchart Perencanaan Jaringan Hotspot Berbayar

Berdasarkan diagram alir pada proyek akhir ini, platform yang digunakan untuk membangun sebuah VM (*virtual machine*) adalah *Google Cloud Platform* dan teknologi yang digunakan untuk membangun layanan *hotspot* berbayar adalah *Userman*. Sistem operasi yang digunakan pada *server* menggunakan OS *Ubuntu* dan *router* yang dibangun pada *Google Cloud Platform* adalah *Mikrotik CHR*. *Userman* digunakan sebagai pusat untuk membuat *voucher*, mengatur *bandwidth* yang akan diberikan pada pengguna *hotspot*. Konfigurasi yang diterapkan pada VM *router* adalah SFTP, *Radius Server* dan PCQ.



Gambar 3. FlowChart Perancangan Sistem Monitoring

Pada Gambar 3.3 di atas menunjukkan *flowchart* untuk perancangan sistem monitor perangkat jaringan, protokol yang digunakan untuk melakukan *monitoring* adalah SNMP dan teknologi yang digunakan untuk melakukan *monitoring* adalah Zabbix Server dan Grafana. SNMP bekerja dengan menerima dan mengirim informasi data pada tiap perangkat jaringan. Zabbix Server digunakan sebagai *server monitoring* yang akan menerima data pada setiap perangkat. Setelah data terkumpul pada Zabbix Server, data akan diolah di Grafana agar dapat memudahkan administrator jaringan dalam melakukan *monitoring* perangkat jaringan.

Pada proses membangun sistem jaringan *hotspot* berbasis *voucher* dikelompokkan menjadi dua bagian, yaitu penggunaan perangkat keras, penggunaan perangkat *virtual machine* dan perangkat lunak. Berikut merupakan penggunaan infrastruktur untuk penelitian proyek akhir, yaitu sebagai berikut:

#### 1. Perangkat Keras

Berikut ini merupakan perangkat keras yang dibutuhkan untuk penelitian proyek akhir dapat dilihat pada tabel 1.

**Tabel 5.** Spesifikasi Router Board.

CPU	CPU QCA9533 650MHz
Size of RAM	32 MB
Storage Size	16 MB
Operating System	Router OS 6.47
Lan Port	3
License	Level 4

#### 2. Perangkat Virtual Machine

Berikut merupakan perangkat *virtual machine* yang dibutuhkan untuk penelitian proyek akhir ini adalah sebagai berikut.

**Tabel 2.** Spesifikasi Server

Jenis	Server
Processor	Intel Skylake
RAM	3.75GB
Disk	30 GB
OS	Ubuntu 18.04 LTS
Machine Type	N1-Standard-1 (1vCPU,3.75GB Memory)

**Tabel 3.** Spesifikasi Router Cloud

Jenis	Router Cloud
Processor	Intel Skylake
RAM	0.6GB
Disk	1 GB
OS	Mikrotik Router OS v6.47
Machine Type	F1-micro (1 vCPU, 0.6 GB memory)

#### 3. Perangkat Lunak

Berikut ini merupakan perangkat lunak yang digunakan untuk penelitian proyek akhir ini adalah sebagai berikut:

- Winbox sebagai perangkat lunak untuk melakukan konfigurasi pada *router*.
- Zabbix Server sebagai *server monitoring* perangkat jaringan.
- Grafana sebagai perangkat lunak untuk membaca dan mengolah data *matrcis* yang telah diberikan oleh *source* menjadi grafik atau data tertulis.

Pada proses implementasi dibutuhkan pemasangan *port service* pada saat sesudah membangun sebuah VM sehingga pengoprasian layanan-layanan yang digunakan dapat berjalan dengan baik. Berikut merupakan port yang digunakan untuk implementasi Proyek Akhir:

**Tabel 4.** Port Service yang digunakan

Port	Keterangan
8291	Mengakses Winbox
443	Menjalankan SSL (Secure Server Layer)
80	Mengkoneksikan Web Server
21	Mengkoneksikan FTP Server
22	Mengaktifkan SSH
1812	Autentikasi dan otorisasi Radius
1813	Penghitungan Radius
3799	Radius Dynamic Authorization
3000	Untuk mengakses <i>web interface</i> Grafana
161	Port SNMP
162	Port SNMP TRAP

Pada proses implementasi agar tiap *router* yang terdapat pada lokasi yang berbeda beda dapat saling terhubung satu sama lain, maka dibutuhkan konfigurasi agar perangkat *router board* dapat terhubung dengan *router cloud* dan menjadi satu jaringan. Berikut ini merupakan konfigurasi pada *router cloud* dan *router board*.

#### 1. Router Cloud

##### Konfigurasi IP

IP Address : 10.184.0.5

Interface : ether1

##### Konfigurasi SSTP Server

Enabled : yes

Max MTU : 1480

Max MRU : 1480

##### Konfigurasi Secret Router Board 1

Name : lokasi1

Local Address : 10.10.10.1

Remote Address : 10.10.10.2

##### Konfigurasi Secret Router Board 2

Name : lokasi2

Local Address : 10.10.11.1

Remote Address : 10.10.11.2

#### 2. Mikrotik Lokasi 1

##### Konfigurasi IP

IP Address (Wlan 1): 199.18.10.8/28

IP Address (Wlan 2): 192.168.1.254

##### Konfigurasi SSTP Client Router Board 1

Connect To : 34.101.92.121

User : lokasi1

#### 3. Mikrotik Lokasi 2

##### Konfigurasi IP

IP Address (ether 1) : 110.136.203.44

IP Address (wlan 2) : 192.168.1.252/24

##### Konfigurasi SSTP Client Router Board 2

Connect To : 34.101.92.121

User : lokasi 2

Pada proses implementasi untuk mengatur dalam proses penanganan otentikasi *user*, otorisasi untuk servis-servis dan perhitungan nilai servis yang dilakukan konfigurasi *Radius Server* sehingga dapat memudahkan administrator jaringan dalam manajemen penggunaan *hotspot* untuk klien. Berikut ini merupakan konfigurasi *Radius Server* pada *router* lokasi 1 dan *router* Lokasi 2.

Hotspot : yes

Address (*secret router* Lokasi 1) : 10.10.10.1

Address (*secret router* Lokasi 2) : 10.10.11.1

Pada proses implementasi untuk manajemen penggunaan *bandwidth* yang digunakan oleh klien pada jaringan *hotspot* dilakukan konfigurasi PCQ sehingga dalam penggunaan *bandwidth* oleh tiap user memperoleh *bandwidth* secara merata. Berikut ini merupakan konfigurasi PCQ pada *router* lokasi 1 dan 2.

Name	Parent	Packet	Limit At (b...)	Max Limit	Avg R...	Queued Bytes	Bytes	Packets
upload	indhome				45.8 kb...	0 B	476.9 ...	6 725
queue1	upload	koneks...		1M	45.8 kb...	0 B	476.9 ...	6 725
download	wlan 1				1002.2 ...	0 B	11.2 MB	8 810
queue2	download	koneks...		1M	960.5 k...	12.2 KB	11.2 MB	8 819

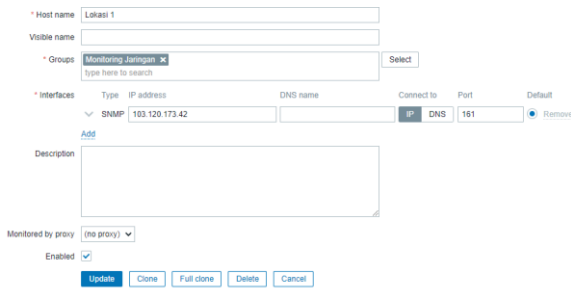
Gambar 4. Konfigurasi PCQ Lokasi 1 dan 2

Pada tahap ini dilakukan instalasi Zabbix Server pada Server Ubuntu sebagai sistem *monitoring*. Kemudian buka terminal dan masukan konfigurasi dibawah ini.

```
# cd /root
# wget
https://repo.zabbix.com/zabbix/4.5/ubuntu/pool/
main/z/zabbix-release/zabbix-release_4.5-
1+bionic_all.deb
dpkg -i zabbix-release_4.5-1+bionic_all.deb
# apt update
# apt install zabbix-server-mysql zabbix-frontend-
php zabbix-apache-conf zabbix-agent
# mysql -uroot -p (masukan password yang di
inginkan)
>mysql> create database zabbix character set utf8
collate utf8_bin;
>mysql> grant all privileges on zabbix.* to
zabbix@localhost identified by 'password';
>mysql> quit;
# zcat /usr/share/doc/zabbix-server-
mysql*/create.sql.gz | mysql -uzabbix -p Zabbix
# nano /etc/zabbix/zabbix_server.conf
(tambahkan konfigurasi)
DBPassword=(password yang telah dibuat)
# nano /etc/zabbix/apache.conf (tambahkan
konfigurasi)
php_value date.timezone Asia/Jakarta
# systemctl restart zabbix-server zabbix-agent
apache2
# systemctl enable zabbix-server zabbix-agent
apache2
Buka Zabbix pada browser dengan ip server
ubuntu (https://34.101.92.121/zabbix)
```

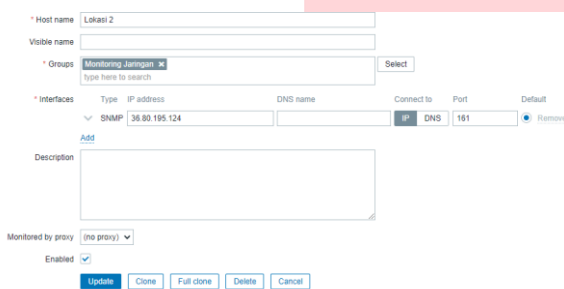
Pada tahap ini untuk melakukan sistem *monitoring* yang akan di lakukan pada perangkat *server*, *router cloud*, dan *router* pada lokasi 1 dan lokasi 2 dilakukan penambahan *host* pada Zabbix Server sehingga perangkat terkoneksi dengan Zabbix Server dapat memberikan data yang kelak akan diolah kembali oleh Zabbix dan di ubah menjadi sebuah informasi sehingga memudahkan administrator jaringan dalam memonitor perangkat jaringan. Berikut ini merupakan konfigurasi Zabbix Server.





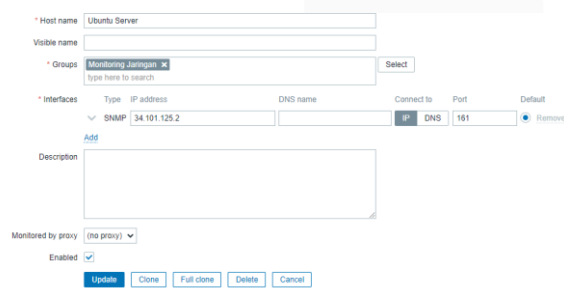
**Gambar 5.** Konfigurasi Zabbix Server Lokasi 1

Pada gambar 5. menunjukkan konfigurasi *host* Zabbix Server untuk lokasi 1 dengan menggunakan *interfaces* SNMP dengan menggunakan ip 103.120.173.42 yang merupakan ip *public* lokasi 1.



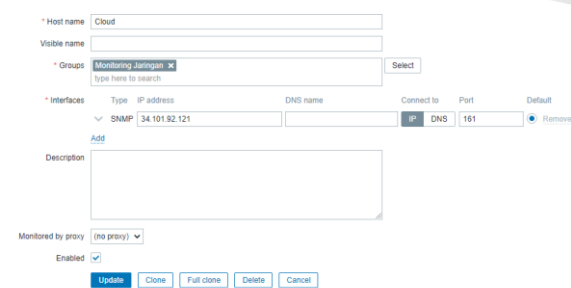
**Gambar 6.** Konfigurasi Zabbix Server Lokasi 2

Pada gambar 6. menunjukkan konfigurasi *host* Zabbix Server untuk lokasi 2 dengan menggunakan *interfaces* SNMP dengan menggunakan ip 36.80.195.124 yang merupakan ip *public* lokasi 2.



**Gambar 7.** Konfigurasi Zabbix Server Ubuntu

Pada gambar 7. menunjukkan konfigurasi *host* Zabbix Server untuk Server Ubuntu dengan menggunakan *interfaces* SNMP dengan menggunakan ip 34.101.125.2 yang merupakan Server Ubuntu.



**Gambar 8.** Konfigurasi Zabbix Router Cloud

Pada gambar 8. menunjukkan konfigurasi *host* Zabbix Server untuk *router cloud* dengan menggunakan *interfaces* SNMP dengan menggunakan ip 34.101.92.121 yang merupakan ip *public router cloud*.

Name	Interface	Availability	Tags	Problems	Status	Latest data	Problems	Graphs	Screens
Cloud	34.101.92.121:161	OK	SNMP	1	Enabled	Latest data	Problems 1	Graphs 7	Screens 1
Lokasi 1	103.120.173.42:161	OK	SNMP	0	Enabled	Latest data	Problems 2	Graphs 11	Screens 1
Lokasi 2	36.80.195.124:161	OK	SNMP	0	Enabled	Latest data	Problems 1	Graphs 11	Screens 1
Ubuntu Server	34.101.125.2:161	OK	SNMP	0	Enabled	Latest data	Problems	Graphs 11	Screens 2
Zabbix server	127.0.0.1:10050	OK	SNMP	0	Enabled	Latest data	Problems	Graphs 19	Screens 3

**Gambar 9.** Daftar Host Zabbix Server

Pada Gambar 9. menunjukkan daftar *host* yang terdeteksi oleh Zabbix Server yang telah terinstalasi pada Server Ubuntu di mana terdapat 4 *host* dan 1 *host default* Zabbix Server yang masing-masing memiliki IP yang berbeda-beda.

Pada tahap ini dilakukan instalasi SNMP pada Server Ubuntu untuk menghubungkan dengan perangkat yang akan dimonitor. Buka terminal dan masukan konfigurasi dibawah ini.

```
# sudo apt update
# sudo apt install snmpd
```

Kemudian dilanjutkan dengan melakukan konfigurasi SNMP Manager Server. Berikut ini merupakan konfigurasi SNMP Manager Server.

```
Masuk pada konfigurasi snmp.conf
$ sudo nano /etc/snmp/snmp.conf
Hapus tanda # seperti di bawah ini
# As the snmp packages come without MIB files
# due to license reasons, loading
# of MIBs is disabled by default. If you added the
# MIBs you can reenable
# loading them by commenting out the following
# line.
mibs :
```

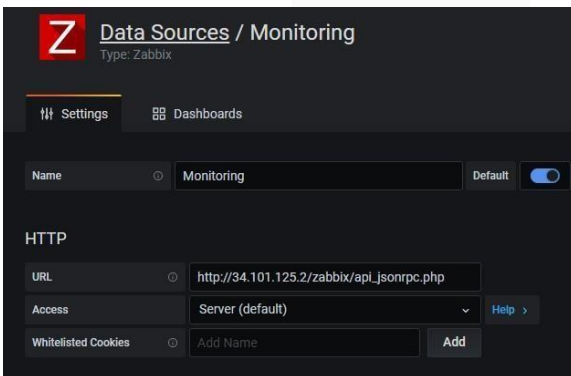
Kemudian dilanjutkan dengan melakukan konfigurasi SNMP Agent Server. Berikut ini merupakan konfigurasi SNMP Agent Server.

```
Masuk pada konfigurasi snmpd.conf
$ sudo nano /etc/snmp/snmpd.conf
Hapus tanda # seperti di bawah ini:
# Listen for connections from the local system
# only
#agentAddress udp:127.0.0.1:161
# Listen for connections on all interfaces (both
# IPv4 *and* IPv6)
agentAddress udp:161,udp6:[::1]:161
Ubah konfigurasi seperti di bawah ini:
rocommunity server localhost
# rocommunity public default -v systemonly
# rocommunity6 public default -v systemonly
Tambahkan IP server:
rocommunity server 34.101.92.121
```

Pada tahap ini dilakukan instalasi Grafana pada Server Ubuntu. Buka terminal dan masukan konfigurasi dibawah ini.

```
# sudo apt update
# sudo wget https://s3-us-west-2.amazonaws.com/grafana-releases/release/grafana_5.2.4_amd64.deb
# sudo apt-get install -y adduser libfontconfig
# sudo dpkg -i grafana_5.2.4_amd64.deb
# sudo nano /etc/apt/sources.list
https://packagecloud.io/grafana/stable/debian/
(copy paste)
# apt install curl
# url https://packagecloud.io/gpg.key | sudo apt-key add -
# sudo apt-get update
# sudo apt-get install grafana
# sudo service grafana-server start
# sudo update-rc.d grafana-server defaults
# sudo systemctl enable grafana-server.service
```

Pada implementasi Proyek Akhir untuk menghubungkan antara Zabbix dengan Grafana dilakukan penambahan data *source* pada Grafana sehingga Grafana dapat terhubung dengan Zabbix Server. Dengan melakukan penambahan data *source* pada Grafana, data yang sudah diolah oleh Zabbix Server dapat dikirimkan ke Grafana dan divisualisaikan sehingga memudahkan administrator jaringan dalam memonitor perangkat pada suatu jaringan. Berikut ini merupakan penambahan data *source* pada Grafana.



**Gambar 10.** Penambahan *Data Sources* Grafana

Pada gambar 10. menunjukkan konfigurasi untuk menghubungkan *source* Zabbix Server dengan memasukan URL Zabbix Server.

Agar *user* dapat mengakses internet pada lokasi *hotspot* yang sudah disediakan dilakukan pembuatan akun *user* dan limitasi *bandwidth* sehingga *user* dapat menggunakan layanan internet kuota dan periode waktu yang berbeda-beda. Adapun tahapan dalam melakukan manajemen *users* dan manajemen *bandwidth*, diantaranya sebagai berikut:

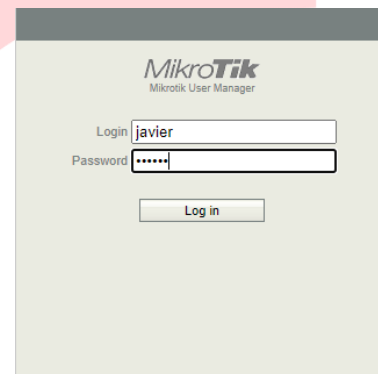
1. Akses *Userman Cloud*
  - a. Login di [http:// 34.101.92.121/userman](http://34.101.92.121/userman)



**Gambar 11.** Akses *Database Userman*

Pada Gambar 11. menunjukkan langkah awal untuk mengakses *userman*.

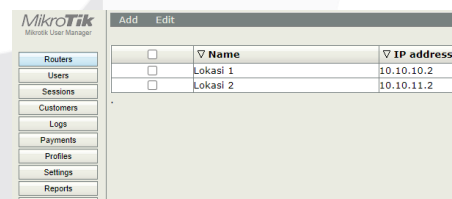
- b. Memasukan *username* dan *password* untuk *login Database Userman Cloud*



**Gambar 12.** Login *Database Userman*

Pada Gambar 12. menunjukkan tampilan *login userman*.

- c. Halaman utama *Database Userman Cloud*

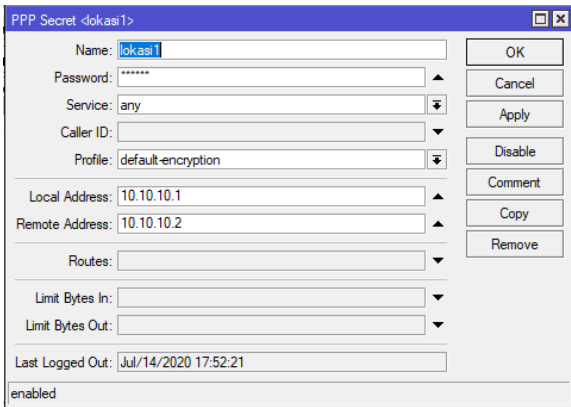


**Gambar 13.** Halaman Utama *Database Userman*

Pada Gambar 13. menunjukkan tampilan awal *menu userman*.

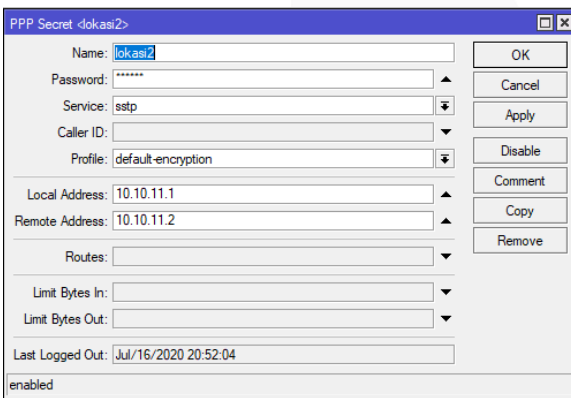
2. Konfigurasi Router

Pada tahap konfigurasi *router* merupakan langkah pertama untuk menambahkan *routers* yang akan terintegrasi dengan *database Userman Cloud*. Adapun langkah- langkah konfigurasinya, klik menu *Routers > Add > New*. Pada tab *Routers Details* masukan parameter-parameter sesuai dengan akun SSTP dari Mikrotik CHR.



Gambar 14. Akun SSTP Lokasi 1

Pada gambar 14. menunjukkan konfigurasi untuk membuat akun SSTP *router* lokasi 1 dengan ip *local address* 10.10.10.1 dan ip *remote address* 10.10.10.2. Kedua ip tersebut kelak akan digunakan untuk menghubungkan antara *userman* dengan *router* lokasi1.

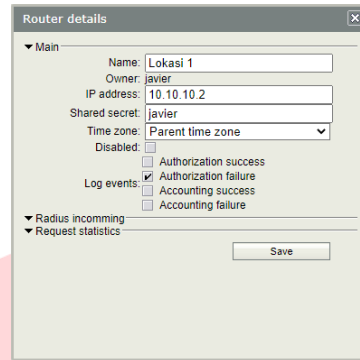


Gambar 15. Akun SSTP Lokasi 2

Pada gambar 15. menunjukkan konfigurasi untuk membuat akun SSTP *router* lokasi 2 dengan ip *local address* 10.10.11.1 dan ip *remote address* 10.10.11.2. Kedua ip tersebut kelak akan digunakan untuk menghubungkan antara *userman* dengan *router* lokasi 2.

a. Router Lokasi 1

- Name : Lokasi 1
- IP Address : 10.10.10.2
- Shared Secret : javier
- Time Zone : Parent Time Zone

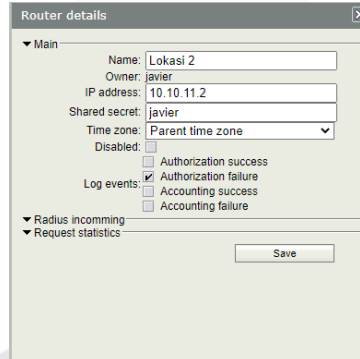


Gambar 16. Konfigurasi Router Lokasi 1

Pada Gambar 16. menunjukkan konfigurasi router lokasi 1 pada *userman*.

b. Router Lokasi 2

- Name : Lokasi 2
- IP Address : 10.10.11.2
- Shared Secret : javier
- Time Zone : Parent Time Zone



Gambar 17. Konfigurasi Router Lokasi 2

Pada Gambar 17. menunjukkan konfigurasi router lokasi 2 pada *userman*.

3. Rancangan Paket Voucher Hotspot

Tabel 5. Paket Voucher

No.	Nama Paket	Speed	Quota	Validity	Harga
1.	Paket 2 GB	10Mb	Unlimited	Unlimited	Rp.2000
2.	Paket 3 GB	10Mb	Unlimited	Unlimited	Rp.3000
3.	Paket 1 Hari	10Mb	Unlimited	1 Hari	Rp.5000
4.	Paket 1 Minggu	10Mb	Unlimited	1 Minggu	Rp.10.000

4. Konfigurasi Limitasi

Tahap ini merupakan upaya pengelolaan limitasi yang akan diberikan ke *user* sesuai rancangan paket *voucher* yang telah dibuat berdasarkan waktu (*time based*), kuota (*quota based*), kecepatan (*rate limit*). Berikut merupakan parameter *commad* untuk melakukan proses konfigurasi limitasi.

**Tabel 6.** Parameter Limitasi

Parameter	Keterangan
Download-limit	kuota limit download (0B: unlimited quota download)
Upload-limit	kuota limit upload (0B: unlimited quota upload)
Transfer-limit	kuota limit download + upload (total)
Uptime	Batasan waktu durasi klien terhubung ke internet
Rate-time- tx	speed limit download (jika kuota sudah habis)
Rate-time-rx	speed limit upload (jika kuota sudah habis)
Min-rate-limit-tx	speed limit download (jika kuota sudah habis)
Min-rate-limit-rx	speed limit upload (jika kuota sudah habis)

c. Limitasi Kuota 1 Hari

**Gambar 20.** Limitasi Kuota 1 Hari

Pada Gambar 20. menunjukan konfigurasi untuk limitasi kuota 1 Hari.

d. Limitasi Kuota 1 Hari

**Gambar 21.** Limitasi Kuota 1 Minggu

Pada Gambar 21. menunjukan konfigurasi untuk limitasi kuota 1 Minggu.

a. Limitasi Kuota 2GB

**Gambar 18.** Limitasi Kuota 2GB

Pada Gambar 18. menunjukan konfigurasi untuk limitasi kuota 2 GB.

b. Limitasi Kuota 3GB

**Gambar 19.** Limitasi Kuota 3GB

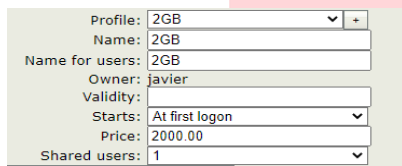
Pada Gambar 19. menunjukan konfigurasi untuk limitasi kuota 3 GB.

5. Konfigurasi *Profile* dan *Limitation*

*Profiles* pada *Userman* merupakan fitur untuk pengelompokan berbagai macam aturan dalam 1 buah *profile* tertentu. Dengan adanya fitur *profiles*, konfigurasi limitasi yang telah dibuat dapat dikelompokkan sesuai dengan klasifikasi kuota dan hari pada *voucher* sehingga memudahkan administrator jaringan dalam pembuatan *user* baru. Berikut merupakan konfigurasi *profiles* dan *limitation*.

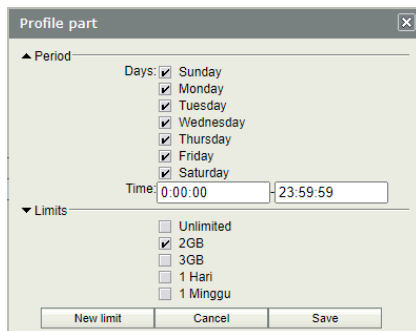
a. Profil Kuota 2GB

- Name : 2GB
- Name for user : 2GB
- Validity : None
- Starts : At first logon
- Price : 2000,00
- Shared Users : 1



Gambar 22. Membuat *Profile* Kuota 2 GB

b. Menghubungkan *Limitation* dengan *profiles* paket kuota 2GB



Gambar 23. Menggabungkan Limitasi *Profile* Kuota 2GB

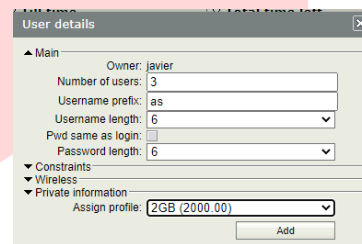
6. *Genrate Username* dan *Password*

Agar pengguna layanan *hotspot* dapat mengakses internet, maka diperlukan sebuah autentikasi *login* untuk setiap pengguna. Autentikasi tersebut berupa *Username* dan *Password* untuk menentukan jenis layanan *hotspot* yang diberikan kepada pengguna dengan ketentuan *userprofile* yang telah dibuat sesuai rancangan paket *voucher*.

Tabel 7. Keterangan Jenis Layanan Hotspot

Number of Users	Jumlah voucher yang akan dibuat
Username prefix	Memberikan huruf awal pada setiap voucher
Username Length	Panjang karakter username
Pwd same as login	Fitur password login akan sama dengan username
Password Length	Panjang karakter password
Assign profile	Jenis paket voucher

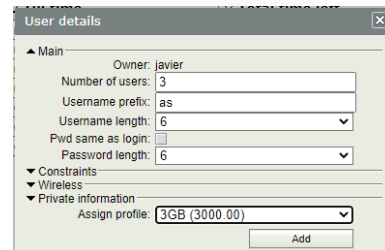
a. *Genrate Paket Voucher* Kuota 2GB



Gambar 24. Menggabungkan Limitasi *Profile* Kuota 2GB

Pada Gambar 24. menunjukkan hasil *genrate* paket *voucher* 2GB

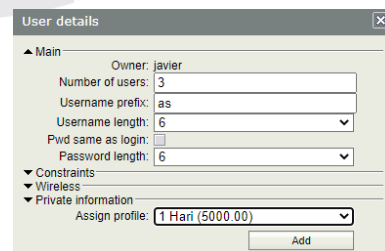
b. *Genrate Paket Voucher* Kuota 3GB



Gambar 25. Menggabungkan Limitasi *Profile* Kuota 3GB

Pada Gambar 25. menunjukkan hasil *genrate* paket *voucher* 3GB.

c. *Genrate Paket Voucher* 1 Hari

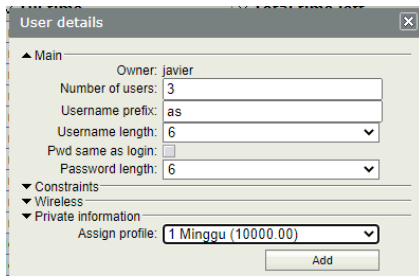


Gambar 26. Menggabungkan Limitasi *Profile* Kuota 1Hari

Pada Gambar 26. menunjukkan hasil *genrate* paket *voucher* 1 hari.



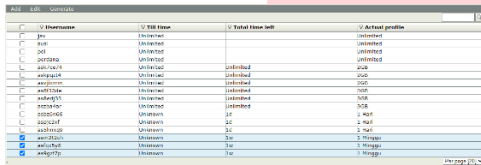
d. Genrate Paket Voucher 1 Minggu



Gambar 27. Genrate Paket Voucher 1 Minggu

Pada Gambar 27. menunjukkan hasil *genrate* paket voucher 1 minggu.

e. List User Data



Gambar 28. List User Data

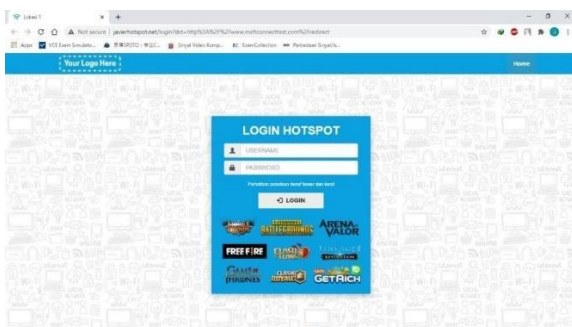
Pada Gambar 28. menunjukkan hasil *user* yang sudah dibuat sebelum digunakan oleh *client*.

V. Hasil dan Analisis

Pada Proyek Akhir ini terdapat beberapa tahapan pengujian yang dilakukan dengan melakukan uji coba *login* pada *hotspot* lokasi 1 kemudian dilanjutkan dengan *login* di *hotspot* lokasi 2 dengan *username* dan *password* yang sama. Pada Proyek Akhir ini telah dipersiapkan 2 buah akun dengan dengan kecepatan *download* dan *upload* 10MB. Kemudian dilanjutkan dengan pengujian sistem monitor perangkat jaringan pada Grafana. Dan tahapan pengujian QOS dilakukan dengan menguji apakah sistem *Queue* PCQ dapat berjalan dengan baik.

1. Pengujian Sistem Autentikasi

Pengujian sistem autentikasi jaringan ini dilakukan untuk mengetahui *user* yang telah disiapkan dapat menggunakan layanan koneksi internet. Pada tahap pengujian autentikasi, dilakukan pengujian *login* pada portal *hotspot* lokasi 1 dan lokasi 2.

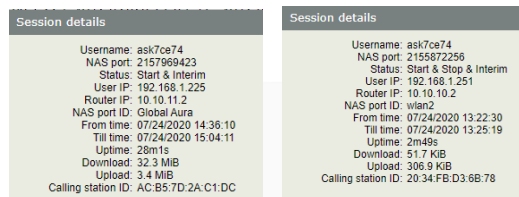


Gambar 29. Halaman Login



Gambar 30. Halaman Login Berhasil

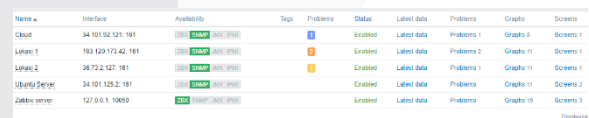
Pada Gambar 30. menunjukkan bahwa *voucher* telah berhasil *login* dan terdapat informasi di mana kuota yang didapat pada *voucher* sebesar 2047.9 MB dan telah melakukan *download* sebesar 18.6KB dan melakukan *upload* sebesar 12.3 KB sesuai dengan konfigurasi limitasi pada *Userman*. Pada gambar 4.3 menunjukkan terjadinya aktivitas *login* pada *hotspot* lokasi 1 dan 2 di mana *login user* berhasil dilakukan pada *hotspot* lokasi 1 dan lokasi 2.



Gambar 31. Halaman Session Database Userman

2. Pengujian Sistem Monitoring Perangkat

Pada pengujian sistem monitor perangkat terdapat beberapa tahapan, tahap pertama melihat apakah perangkat yang telah terkonfigurasi SNMP dapat terhubung dengan *Zabbix Server*, tahap berikutnya mengecek konektivitas antara *Zabbix* dengan *Grafana*. Langkah terakhir mengecek apakah *dashboard* yang dibuat pada *Grafana* telah sesuai dengan data *source Zabbix Server*.



Gambar 32. Daftar perangkat yang terhubung dengan Zabbix Server

Pada Gambar 31. menunjukkan bahwa perangkat *Server*, *Mikrotik CHR*, *Router* Lokasi 1 dan Lokasi 2 telah terhubung dengan *Zabbix Server* melalui protokol SNMP. Apabila salah satu dari perangkat terputus pada jaringan, status SNMP akan menjadi merah dan akan muncul status *problem* yang menginformasikan jika perangkat tidak terhubung dengan *Zabbix Server*.

Latest data

Host	Name	Last check	Last value	Change
liberty Server	CPU (17 items)	2020-07-24 15:58:11	264.549	+15.9162
	Current load/1s per second	2020-07-24 15:58:11	9%	
	CPU guest time	2020-07-24 15:58:11	9%	
	CPU iops time	2020-07-24 15:58:11	91.7513%	-7.7346%
	CPU irq time	2020-07-24 15:58:11	9%	
	CPU kernel time	2020-07-24 15:58:11	3.7483%	+0.1532%
	CPU nice time	2020-07-24 15:58:11	9%	
	CPU softirq time	2020-07-24 15:58:11	9.05226%	+0.61672%
	CPU steal time	2020-07-24 15:58:11	9%	
	CPU system time	2020-07-24 15:58:11	3.4157%	+0.68957%
	CPU user time	2020-07-24 15:58:11	3.962%	-0.95271%
	CPU utilization	2020-07-24 15:58:11	2.287%	+7.7346%
	Interrupts per second	2020-07-24 15:58:11	92.9051	-8.4966

Gambar 33. Latest Data Host Zabbix Server

Pada Gambar 33. merupakan data terbaru yang telah dikirim dari perangkat jaringan melalui protokol SNMP sebelum divisualisasikan ke Grafana.



Gambar 34. Dashboard Monitoring Perangkat Jaringan

Pada Gambar 34. menunjukkan data *source* Zabbix telah terhubung dengan Grafana dan di dalam *dashboard* terdapat informasi perangkat jaringan, di mana menunjukkan bahwa sistem monitor dapat berjalan dengan baik.

### 3. Pengujian Sistem PCQ

Pada tahap ini dilakukan analisis hasil pengukuran QOS (*Quality of Service*) baik sebelum maupun sesudah melakukan manajemen *bandwidth* internet dengan menggunakan metode PCQ (*Per Connection Queue*) terhadap kualitas layanan dan kinerja jaringan *hotspot*. Pengujian dilakukan dengan menggunakan *software Wireshark*. Dalam proyek akhir ini, parameter QOS yang dianalisis yaitu:

- Throughput*, merupakan kecepatan (*rate*) transfer data efektif, yang diukur dalam bps (*bit per second*). *Throughput* adalah jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu.
- Delay*, merupakan total waktu yang dilalui suatu paket dari pengirim ke penerima melalui jaringan.
- Jitter*, merupakan variasi dari *delay end-to-end*.
- Packet Loss*, merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang, dapat terjadi karena *collision* dan *congestion* pada jaringan.

Tabel 8. Hasil Nilai Rata-Rata Parameter Pengujian

Metode	Parameter	Hasil
Tanpa PCQ	Throughput (Mbps)	8.2918
	Packet Loss (%)	1.8198
	Jitter (ms)	0.838202
	Delay (ms)	0.838229
PCQ	Throughput (Mbps)	6.1142
	Packet Loss (%)	1.8677
	Jitter (ms)	1.077625
	Delay (ms)	1.077639

Pada tabel 8 merupakan hasil rata-rata setiap parameter dari keseluruhan pengujian yang dilakukan. Pada sub-bab ini dijelaskan perbandingan antara metode PCQ dan tanpa metode PCQ. Pada metode PCQ parameter pengujian *throughput* sedikit lebih kecil yaitu memperoleh 5.775 Mbps dibanding tanpa metode PCQ yang memperoleh 5.923 Mbps. Hasil ini dikarenakan *throughput* yang di berikan kepada pengguna *hotspot* telah terbagi secara merata. Pada metode PCQ parameter pengujian *Packet Loss* sedikit lebih besar yaitu memperoleh 4.172 % dibanding tanpa metode PCQ yang memperoleh 3.028 %. Pada metode PCQ parameter pengujian *Jitter* sedikit lebih unggul yaitu memperoleh 0.730 ms dibanding tanpa metode PCQ yang memperoleh 1.299 ms. Pada metode PCQ parameter pengujian *Delay* sedikit lebih unggul yaitu memperoleh 0.730 ms dibanding tanpa metode PCQ yang memperoleh 1.299 ms.

## VI. Kesimpulan dan Saran

### 1. Kesimpulan dan Saran

Berdasarkan hasil perancangan, pengujian dan analisis yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut:

- Implementasi jaringan *hotspot* berbayar berbasis *voucher* dapat di aplikasikan dengan menggunakan *Google Cloud Platform*.
- Pada lokasi kedua menggunakan layanan ISP Indihome yang hanya memperoleh IP *Public* dinamik sehingga konektifitas SNMP dengan *Zabbix Server* sering terputus.
- Pada lokasi kedua menggunakan layanan ISP Indihome memiliki ONT yang tidak dapat melakukan ping pada IP *Public* yang di peroleh sehingga tidak dapat menggunakan protokol SNMP, bisa di selesaikan dengan menerapkan *mode bridge* pada ONT ke Mikrotik *Router* sehingga IP *Public* yang diperoleh oleh ISP dapat langsung di terima oleh Mikrotik *Router*.
- Sistem *monitoring* perangkat jaringan dapat dilakukan dengan menggunakan protokol SNMP.
- Sistem *voucher* dapat di aplikasikan pada *Hotspot* Lokasi 1 dan Lokasi 2.
- Sistem PCQ dapat di aplikasikan sehingga *bandwidth* yang digunakan oleh pengguna akses jaringan internet *Hotspot* dapat terbagi secara merata.

## 2. Saran

Berdasarkan hasil pembangunan Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu:

- Menggunakan IP *Public Static* pada setiap lokasi router sehingga IP *Public* tidak akan berubah setiap kali ONT dalam keadaan *restart*.
- Menambahkan perangkat jaringan sehingga dalam pengaplikasiannya dapat lebih kompleks.
- Pada tiap lokasi memiliki jumlah *bandwidth* yang besar sehingga dalam pembagian *bandwidth* dapat lebih mudah.

## REFERENSI

- Y. A. Nugroho, *Perancangan dan Impelementasi Centralized User Manager di Cloud*, Yogyakarta: Universitas Sanata Dharma, vol. 23, no. 3. Yogyakarta, 2019.
- F. BAKRI, *MEMBANGUN JARINGAN HOTSPOT MENGGUNAKAN SISTEM VOUCHER*. Bandung: Universitas Telkom, 2017.
- M. U. Farouk, M. Ramdhani, and T. A. Wibowo, "Implementasi jaringan hotspot berbasis voucher dan billing menggunakan mikrotik router os," *Implementasi Jar. hotspot Berbas. voucher dan billing menggunakan mikrotik router os*, 2012.
- C. Kurniawan, "Perancangan Jaringan Hotspot dengan Sistem Voucher Menggunakan Mikrotik pada Jaringan RT / RW Net," no. September, 2014.
- W. F. Pattipeilohy, "Analisis dan Perancangan User Manager pada Mikrotik Router dengan Sistem Pembelian Kredit Voucher," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 5, no. 1, p. 64, 2016, doi: 10.32736/sisfokom.v5i1.30.
- A. N. Asyikin, N. Saputera, and E. Yohanes, "Sistem Manajemen Hotspot Di Politeknik Negeri Banjarmasin Menggunakan Mikrotik Router OS," *J. POROS Tek.*, vol. 5, no. 1, pp. 31–35, 2013.
- A. Zainuddin, L. Affandi, and A. D. Susilo, "Honeypot Dan Ids Di Kampus Stmik Ppkia Pradnya Paramita Malang," *J. Teknol. Inf.*, vol. 5, no. 2, 2014.
- L. Christiani, "Peluang dan Tantangan Penerapan Cloud Computing (Komputasi Awan) Sebagai Solusi Automasi Kerjasama Antar Perpustakaan," *Anuva*, vol. 2, no. 1, p. 43, 2018, doi: 10.14710/anuva.2.1.43-53.
- M. I. P. RISKI, "ANALISA AKSESIBILITAS JARINGAN INTRANET DENGAN MENGGUNAKAN L2TP/IPSEC MENGGUNAKAN VIRTUAL PRIVATE NETWORK," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.
- Desmulyati and M. R. P. Putra, "Load Balance Design of Google Cloud Compute Engine VPS with Round Robin Method in PT. Lintas Data Indonesia," *Sinkron*, vol. 3, no. 2, p. 147, 2019, doi: 10.33395/sinkron.v3i2.10064.
- D. Wijonarko, "Zabbix Network Monitoring Sebagai Perangkat Monitoring Jaringan Di Skpd Kota Malang," pp. 27–38.
- R. M. Febriana, "IMPLEMENTASI SISTEM MONITORING MENGGUNAKAN PROMETHEUS DAN GRAFANA," *Semin. Nas. Telekomun. dan Inform. (SELISIK 2016)*, vol. 13, no. 1, pp. 164–169, 2020, [Online]. Available: <http://jitter.widyatama.ac.id/index.php/Selisik2016/article/download/126/103>.
- Mirsantoso, T. U. Kalsum, and R. Supardi, "Implementasi dan Analisa Per Connection Queue (PCQ) Sebagai Kontrol Penggunaan Internet Pada Laboratorium Komputer," *J. Media Infotama*, vol. 11, no. 2, pp. 139–148, 2015, [Online]. Available: <https://jurnal.unived.ac.id/index.php/jmi/article/view/261>.
- C. J. M. Fedor, M. Schoffstall, and J. Davin, "A Simple Network Management Protocol (SNMP)," *J. Chem. Inf. Model.*, pp. 1–36, 1990.
- D. Harrington, R. Presuhn, and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks Status," *New York*, vol. 133, no. 57, pp. 1–5, 2002.
- K. McCloghrie and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II," *J. Chem. Inf. Model.*, pp. 1–70, 1991.
- Diana and F. Maulana, "Implementasi Simple Network Management Protocol (Snmp) Pada Aplikasi Monitoring Jaringan Berbasis Website(Studi Kasus Universitas Muhammadiyah Bengkulu)," *J. Inform.*, vol. 16, no. 2, pp. 126–135, 2016.
- H. Kusbandono and E. M. Syafitri, "Penerapan Quality Of Service (QoS) dengan Metode PCQ untuk Manajemen Bandwidth Internet pada WLAN Politeknik Negeri Madiun," *Res. Comput. Inf. Syst. Technol. Manag.*, vol. 2, no. 1, p. 7, 2019, doi: 10.25273/research.v2i1.3743.
- E. Febriyanti, S. Raharjo, and M. Sholeh, "Perbandingan Manajemen Bandwidth Menggunakan Metode FIFO (First In First Out) dan PCQ (Peer Connection Queue) Pada Router Mikrotik (Studi Kasus Pada Laboratorium Komputer Jaringan, Institut Sains & Teknologi AKPRIND Yogyakarta)," *J. JARKOM*, vol. 5, no. 2, pp. 89–98, 2017.