



## IMPLEMENTASI MANAGEMENT BANDWIDTH USER BERBASIS VPS DENGAN API MANAGE SERVICES

Kikim Mukiman<sup>1</sup>, Subnadri<sup>2</sup>, Andi Cahyono<sup>3</sup>, Iwan Mulyana<sup>4</sup>, Ramdani<sup>5</sup>

<sup>1</sup>Universitas Bani Saleh, [kikim@ubs.ac.id](mailto:kikim@ubs.ac.id)

<sup>2</sup>Universitas Bani Saleh, [subandri@ubs.ac.id](mailto:subandri@ubs.ac.id)

<sup>3</sup>Universitas Bani Saleh, [andiiwrite@gmail.com](mailto:andiiwrite@gmail.com)

<sup>4</sup>STMIK Al-Muslim, [iwan.mulyana@almuslim.ac.id](mailto:iwan.mulyana@almuslim.ac.id)

<sup>5</sup>Universitas Bani Saleh, [ramdani@ubs.ac.id](mailto:ramdani@ubs.ac.id)

### ABSTRAK

Pengelolaan bandwidth yang efektif menjadi krusial dalam menjaga kualitas layanan jaringan dan menjamin pemerataan akses bagi semua pengguna. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah sistem manajemen bandwidth pengguna yang terpusat dan terotomasi. Sistem dibangun dengan memanfaatkan Virtual Private Server (VPS) sebagai infrastruktur yang andal dan fleksibel, serta Application Programming Interface (API) Management Services untuk menyediakan antarmuka pemrograman aplikasi yang terkelola dengan baik. Metode implementasi meliputi tahap analisis kebutuhan, perancangan arsitektur, konfigurasi kebijakan bandwidth pada VPS, pengembangan API, serta pengujian fungsional dan performa. Hasil implementasi menunjukkan bahwa sistem yang dibangun berhasil membatasi bandwidth pengguna secara real-time berdasarkan kebijakan yang telah ditetapkan, seperti pembatasan kecepatan download dan upload. Keunggulan solusi ini terletak pada sentralisasi kontrol, yang memungkinkan administrasi bandwidth untuk banyak pengguna atau perangkat dilakukan melalui panggilan API tunggal, sehingga meningkatkan efisiensi operasional. Sistem ini memberikan solusi yang skalabel dan terukur untuk manajemen bandwidth, khususnya dalam lingkungan seperti penyedia layanan internet kecil, warnet, atau jaringan kampus.

**Kata Kunci:** Manajemen Bandwidth, VPS, API, Quality of Service (QoS), Jaringan Komputer, Layanan Terkelola.

### ABSTRACT

*Effective bandwidth management is crucial for maintaining network service quality and ensuring equitable access for all users. This research aims to design and implement a centralized and automated user bandwidth management system. The system is built using a Virtual Private Server (VPS) as a reliable and flexible infrastructure, along with API Management Services to provide a well-managed application programming interface. The implementation method includes stages of needs analysis, architecture design, configuration of bandwidth policies on the VPS, API development, and functional and performance testing. The implementation results show that the developed system successfully limits user bandwidth in real-time based on predefined policies, such as download and upload speed throttling. The advantage of this solution lies in its centralized control, which allows bandwidth administration for multiple users or devices to be handled through a single API call, thereby enhancing operational efficiency. This system provides a scalable and measurable solution for bandwidth management, particularly in environments such as small internet service providers, internet cafes, or campus networks.*

**Keywords:** Bandwidth Management, VPS, API, Quality of Service (QoS), Computer Network, Managed Services.



## PENDAHULUAN

Perkembangan teknologi digital dan meningkatnya ketergantungan terhadap koneksi internet telah menciptakan tantangan besar dalam pengelolaan sumber daya jaringan. Bandwidth, sebagai salah satu aset paling kritis, seringkali menjadi sumber permasalahan, mulai dari kelambatan akses, ketidakstabilan koneksi, hingga ketidakadilan dalam distribusi sumber daya antar pengguna. Pada lingkungan jaringan skala menengah seperti warnet, kampus, perkantoran, atau penyedia layanan internet (ISP) kecil, permintaan akan bandwidth yang terbatas seringkali tidak sebanding dengan jumlah pengguna yang terus bertambah. Tanpa adanya pengelolaan yang efektif, pengguna yang memanfaatkan bandwidth secara intensif (seperti untuk streaming video high-definition atau mengunduh file besar) dapat secara tidak sadar mengganggu pengalaman pengguna lain yang membutuhkan akses untuk aktivitas penting.

Solusi tradisional untuk manajemen bandwidth seringkali bergantung pada perangkat keras (hardware) khusus seperti router yang dilengkapi fitur Quality of Service (QoS) atau perangkat firewall. Meskipun efektif, solusi berbasis perangkat keras ini umumnya memerlukan investasi awal yang signifikan dan konfigurasi yang kompleks, sehingga kurang fleksibel dan terjangkau untuk skala yang lebih kecil. Di sisi lain, solusi berbasis perangkat lunak (software) yang dijalankan pada Virtual Private Server (VPS) menawarkan alternatif yang lebih ekonomis dan skalabel. VPS memberikan kontrol penuh atas lingkungan server dengan biaya yang relatif terjangkau, memungkinkan implementasi kebijakan jaringan yang spesifik sesuai kebutuhan.

Namun, mengelola kebijakan bandwidth secara manual pada VPS untuk setiap pengguna atau perangkat masih merupakan tugas yang memakan waktu dan rentan terhadap human error. Oleh karena itu, diperlukan sebuah lapisan automasi dan sentralisasi. Di sinilah peran Application Programming Interface (API) Management Services menjadi sangat krusial. Dengan mengintegrasikan API, sistem manajemen bandwidth dapat dikendalikan secara terpusat dan terprogram. Administrator tidak perlu lagi masuk secara langsung ke dalam server VPS untuk mengubah konfigurasi; segala perubahan seperti penambahan kuota, pembatasan kecepatan, atau pemantauan penggunaan dapat dilakukan melalui panggilan API yang sederhana dari sebuah dashboard atau aplikasi terintegrasi.

Berdasarkan latar belakang tersebut, penelitian ini dilakukan untuk mengimplementasikan sebuah sistem **Management Bandwidth User berbasis VPS yang dikendalikan melalui API Management Services**. Solusi ini diharapkan dapat menjawab tantangan efisiensi, sentralisasi kontrol, dan skalabilitas dalam mengelola bandwidth bagi banyak pengguna..

## TINJAUAN PUSTAKA

### MANAJEMEN AKSES INTERNET BERBASIS LIMITASI MENGGUNAKAN MIKHMON DAN ROUTER MIKROTIK

Jurnal referensi dari Erni Krisnaningsih, Saleh Dwiayatno, Ahmad Dedi Jubaedi, Yandri atah Maulana, Suhardianto, JURNAL PROSISKO, Vol 10 No 2 September 2023, E-ISSN: 2597-9922 dan P-ISSN: 2406-7733 , dengan judul “MANAJEMEN AKSES INTERNET BERBASIS LIMITASI MENGGUNAKAN MIKHMON DAN ROUTER MIKROTIK” dilakukan penelitian ini untuk menciptakan sistem manajemen akses internet yang efektif dan stabil di rumah. Penelitian menerapkan Manajemen Akses Internet Berbasis Limitasi menggunakan Mikhmon dan Router Mikrotik. Tujuannya adalah membatasi penggunaan waktu internet dan membagi bandwidth secara merata untuk setiap pengguna, meningkatkan

kecepatan dan stabilitas jaringan di rumah. Penelitian berlangsung dari Oktober hingga Maret 2021/2020 dengan metode observasi. Hasilnya adalah sistem limitasi

yang meningkatkan efektivitas, kenyamanan, dan keamanan penggunaan internet di rumah. Fitur manajemen hotspot memungkinkan pembagian bandwidth yang merata dan pembatasan waktu internet untuk setiap pengguna. Implementasi melibatkan satu Modem ISP sebagai sumber internet, satu Routerboard Mikrotik sebagai manajemen jaringan dan router gateway, serta satu access point untuk penyebaran jaringan melalui LAN dan WLAN..

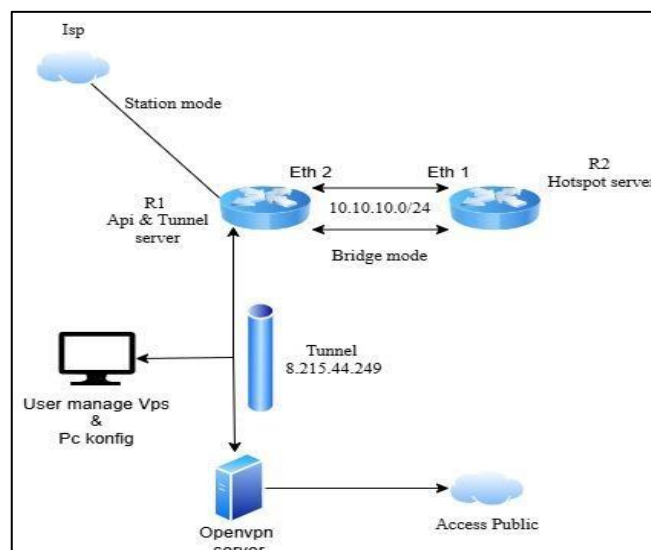
## **NETWORK OPTIMIZATION AND NETWORK USER ACCESS LIMITATIONS USING MIKHMOM SERVER AT DISKOMINFO PALEMBANG CITY**

Jurnal referensi dari Guntoro Barovih, Lisa Artika, Fahmi Ajis Manto, LC INTERNATIONAL JOURNAL OF STEM, Vol 3 Desember 2022, E-ISSN: 2708-7123, dengan judul “ **NETWORK OPTIMIZATION AND NETWORK USER ACCESS LIMITATIONS USING MIKHMOM SERVER AT DISKOMINFO PALEMBANG CITY**” dilakukan penelitian ini untuk membuat sistem yang menggunakan mekanisme otorisasi guna membatasi pengguna jaringan. Penelitian ini menggabungkan berbagai sistem kerja antara router Mikrotik sebagai gateway untuk tamu yang masuk ke jaringan di DISKOMINFO Kota Palembang dan server Mikhmon. Teknik yang digunakan adalah Network Development Life Cycle (NDLC). Sistem ini memungkinkan pemberian akses internet dengan kode voucher yang memiliki batas waktu; sehingga pengguna akan dibatasi ketika waktu penggunaan telah habis. Hasil penelitian menunjukkan bahwa teknologi ini dapat membantu DISKOMINFO Kota Palembang memisahkan jaringan internal dari jaringan tamu; menciptakan jaringan yang aman meskipun tamu mengakses internet melalui jaringan internal perusahaan.

## **METODE PENELITIAN**

### **Topologi Jaringan**

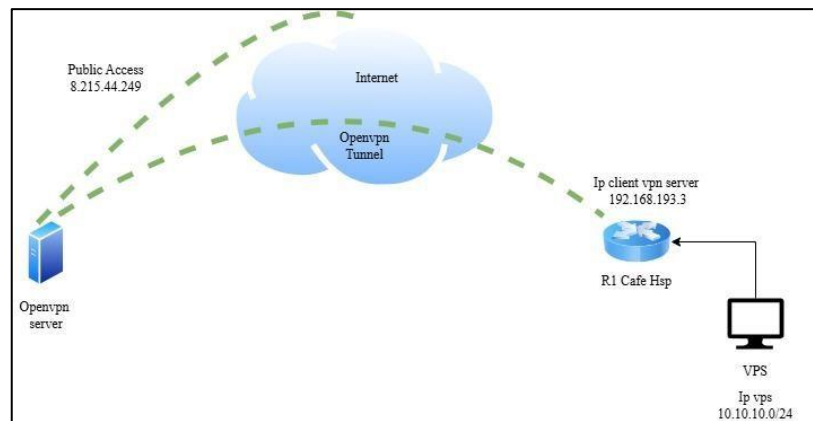
Pada tahap akan membuat desain topologi jaringan yang nantinya akan digunakan sebagai tumpuan sebelum melakukan tahap implementasi. Pada dasarnya topologi jaringan biasanya berisi tentang informasi dari alur dan informasi pada alamat ip yang akan digunakan.



## Topologi Jaringan

### Desain Virtual Private Server Dengan Tunnelling Openvpn

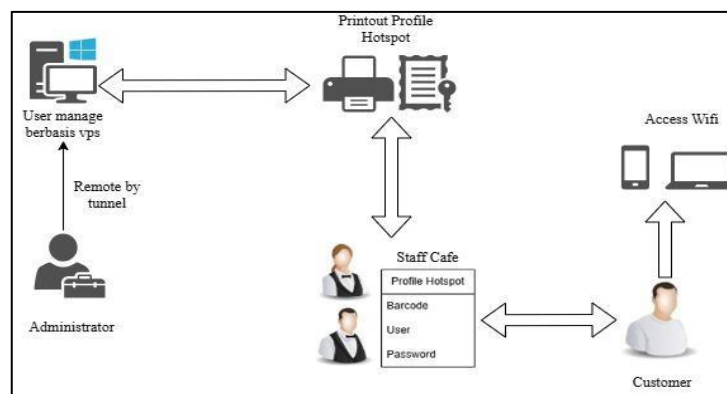
Setelah selesai membuat topologi jaringan, selanjutnya akan membuat desain dari Virtual Private Server (VPS) yang nantinya akan dijalankan dengan menggunakan *tunneling* OpenVPN. Desain ini memastikan bahwa VPS dapat diakses dengan aman dari berbagai jaringan, memberikan fleksibilitas dan aksesibilitas tinggi bagi administrator untuk mengelola sistem dari jarak jauh.



### Topologi Tunneling Openvpn

#### Desain Client

Berikutnya, mendesain alur kerja sistem bagi pengunjung untuk bisa mengakses *Wifi* seperti pada gambar di bawah.



### Topologi Alur Kerja Client

#### Konfigurasi Identitas Pada Router 1 Dan 2

Konfigurasi identitas pada Router 1 dan 2 untuk memastikan setiap router memiliki nama yang unik dan tidak tertukar saat melakukan konfigurasi.

```
Do you want to see the software license? [Y/n]: n
[?]          Gives the list of available commands
command [?]  Gives help on the command and list of arguments

[Tab]        Completes the command/word. If the input is ambiguous,
              a second [Tab] gives possible options

/            Move up to base level
..           Move up one level
/command     Use command at the base level

[admin@MikroTik] >
[admin@MikroTik] >
[admin@MikroTik] > system identity
edit export get print set
[admin@MikroTik] > system identity set name=Router-Server
[admin@Router-Server] >
[admin@Router-Server] >
[admin@Router-Server] >
```

### Konfigurasi Identitas Router 1

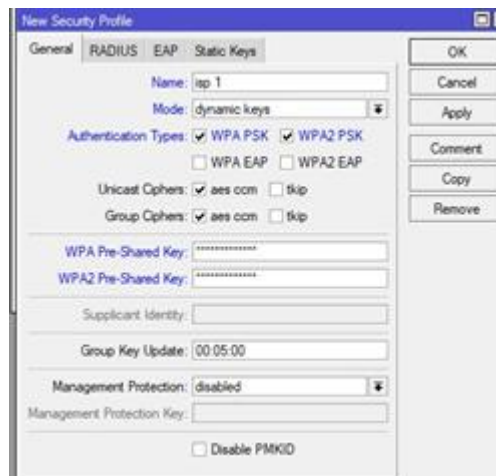
### Konfigurasi Bridge Untuk Konektivitas Antara Router 1 Dan 2

Konfigurasi bridge pada router server untuk memungkinkan konektivitas antara Router 1 dan 2.

### Konfigurasi Akses Internet Pada Router Server

Pada router server akses internet yang digunakan mengambil dari isp atau provider menggunakan wlan 1 dengan mode station.

### Konfigurasi Security Profile ISP:



### Konfigurasi security profile

### Sambungkan ISP pada *Interface* wlan1:

Konfigurasi untuk mendapatkan IP address dari ISP: Buka terminal dan lakukan konfigurasi untuk mendapatkan IP address dari ISP atau provider.

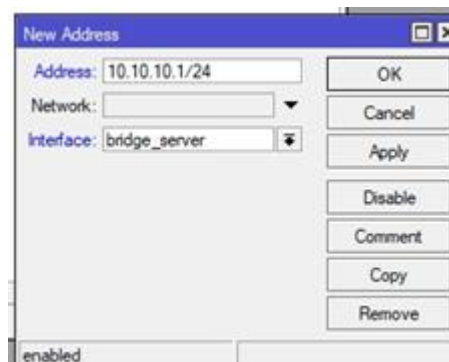
```
[admin@Router-Server] > ip dhcp-client add
interface: wlan1
[admin@Router-Server] > ip dhcp-client enable
numbers
[admin@Router-Server] > ip dhcp-client enable 0
[admin@Router-Server] > ip dhcp-client pr
Flags: X - disabled, I - invalid, D - dynamic
#  INTERFACE  USE-PEER-DNS  ADD-DEFAULT-ROUTE  STATUS  ADDRESS
0  wlan1      yes           yes               bound   192.168.0.108/24
[admin@Router-Server] >
```

Dhcp client router 1

Tes koneksi internet: Lakukan *ping* ke beberapa website untuk menguji koneksi internet.

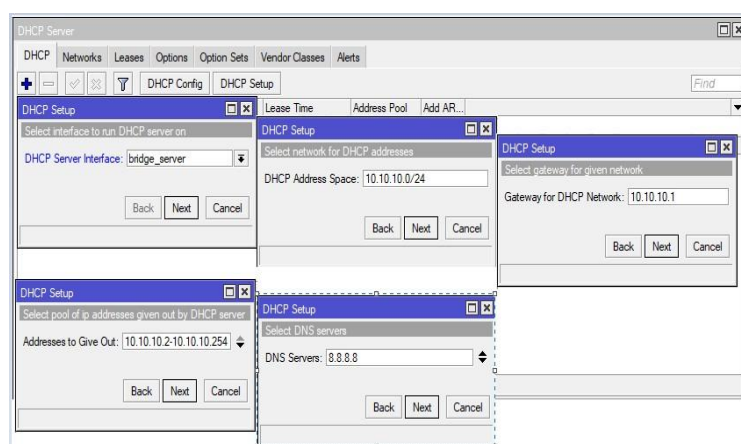
### Konfigurasi Dhcp Server

Lakukan konfigurasi DHCP server untuk memberikan akses internet kepada VPS dan router hotspot. Tambahkan IP address pada *Interface bridge\_server*: Konfigurasi IP address sesuai dengan tabel IP pada topologi yang telah dibuat,



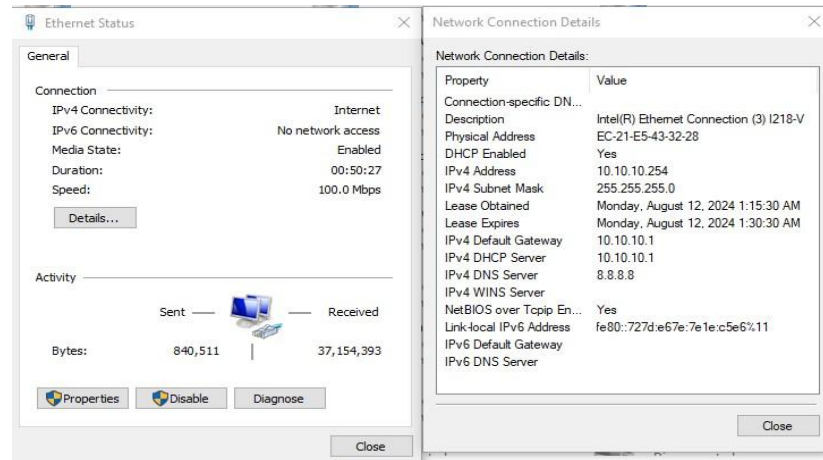
Tambah ip interface bridge\_server

### Konfigurasi DHCP server:



Tambah dhcp server router 1

Tes client pada DHCP server: Pastikan VPS yang terhubung mendapatkan IP secara dinamis sesuai dengan konfigurasi DHCP server yang telah dibuat.



## Hubungan vps ke jaringan

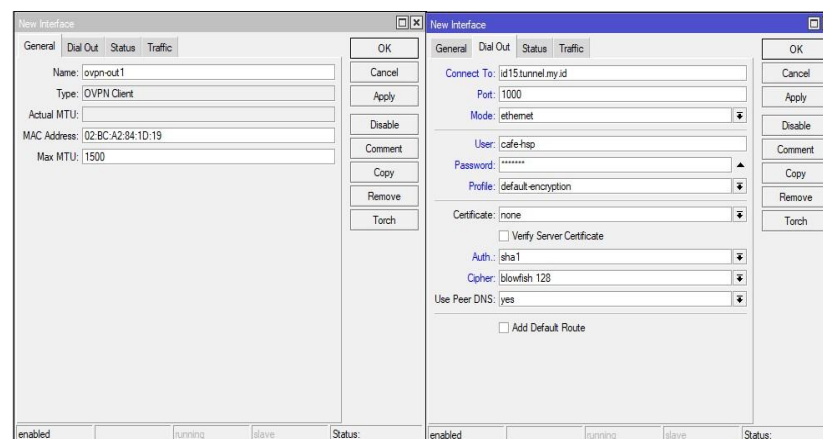
### Mengaktifkan Port Api Di Mikrotik

Langkah pertama adalah mengaktifkan port API pada MikroTik. Port API ini akan digunakan untuk menghubungkan MikroTik dengan User Management berbasis VPS. Aktivasi port ini memungkinkan komunikasi antara router dan VPS sehingga manajemen pengguna dan pengaturan lainnya dapat dilakukan secara remote melalui API. Pastikan port yang digunakan sesuai dengan yang dikonfigurasi di VPS untuk menghindari konflik atau kesalahan dalam pengoperasian. Aktifkan service api dan *WWW* di mikrotik:

### Konfigurasi *Tunneling* Openvpn Pada Vps

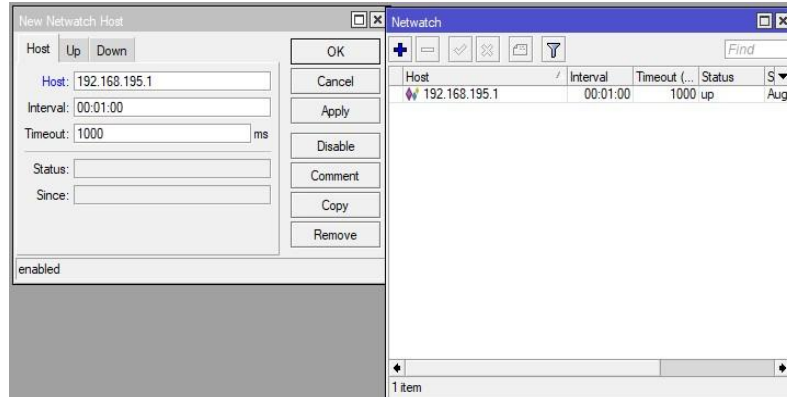
Selanjutnya, dilakukan konfigurasi *tunneling* menggunakan OpenVPN pada VPS. Tujuan dari konfigurasi ini adalah untuk memungkinkan VPS diakses secara publik melalui jaringan internet. Dengan menempatkan VPS pada server OpenVPN, VPS tersebut dapat diakses dari berbagai lokasi dan jaringan, sehingga memberikan fleksibilitas bagi administrator untuk mengelola sistem secara jarak jauh. Proses ini melibatkan pengaturan server OpenVPN pada VPS dan pengaturan client pada router MikroTik agar dapat terhubung dengan server tersebut.

Tambahkan openvpn client dan isi sesuai server vpn.



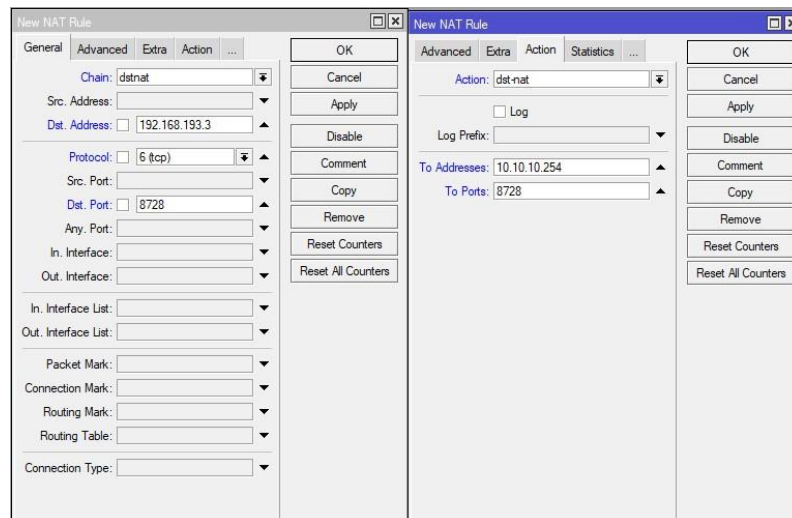
### Isi open vpn client

Tambahkan ip netwatch untuk cek server tunnel :



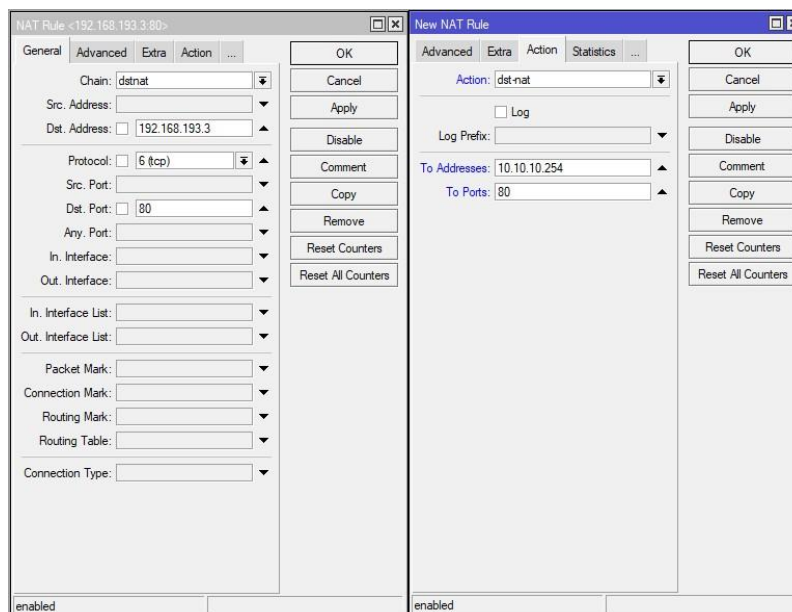
Tambah ip netwatch

Konfigurasi setingan firewall ke arah port api (8728) :



Setingan nat port api

Konfigurasi setingan firewall ke arah port *WWW* (80) :





## Setingan nat port *WWW*

### Konfigurasi Hotspot Server Pada Router 2

Setelah *tunneling* dikonfigurasi, langkah berikutnya adalah mengatur hotspot server pada Router 2. Router ini akan bertindak sebagai server hotspot yang mengelola koneksi pengguna ke jaringan Wi-Fi.

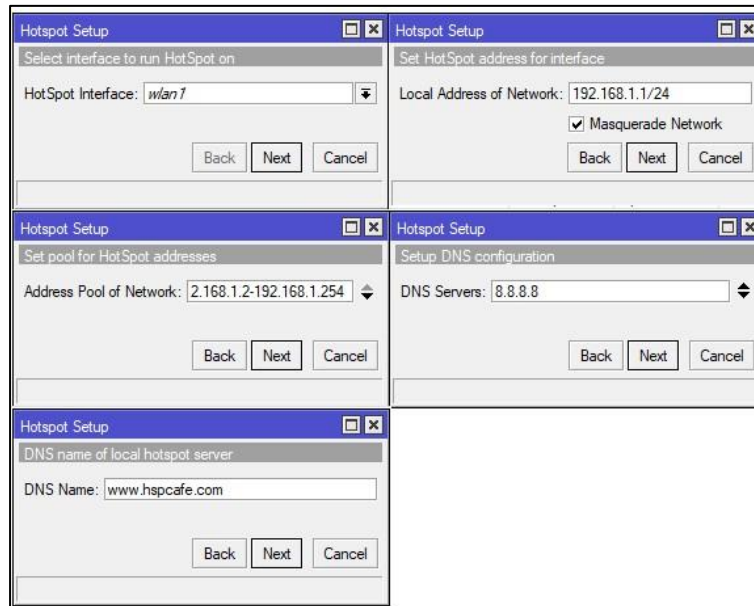
Konfigurasi interface wlan1 kedalam mode ap bridge :

### Mode ap bridge

Konfigurasi Ip address pada wlan1 :

### Ip pada wlan1

Konfigurasi Hotspot server di router 2 :



Setup hotspot server

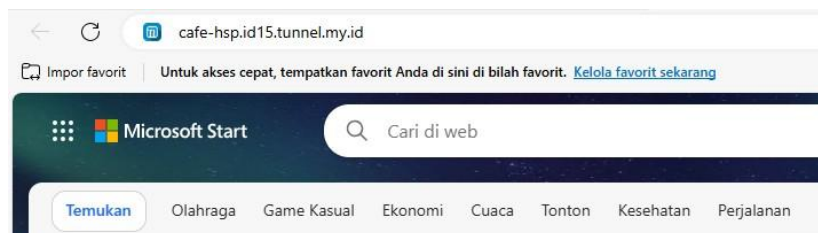
## HASIL DAN PEMBAHASAN

### Hasil

#### Menghubungkan Hotspot Server Di User Manage Berbasis Vps

Bagian ini adalah menyambungkan hotspot server yang telah dikonfigurasi ke dalam sistem User Management berbasis VPS. Proses ini melibatkan pengaturan URL API, kredensial autentikasi, serta pengujian konektivitas antara router dan VPS. Setelah tersambung, hotspot server akan menggunakan VPS untuk autentikasi pengguna, pemantauan, dan pengelolaan jaringan secara keseluruhan.

Jalankan domain *tunneling* yang telah dikonfigurasi pada browser.



#### User manage pada web

Setelah masuk ke web, kemudian login dengan user dan password default mikhmon.

Mengubah info login pada web user manage.



### Mengubah akun user manage

Pada laman kiri klik Add Router untuk menghubungkan hotspot server pada router ke user manage berbasis api melalui api mikrotik.

### Add router pada user manage

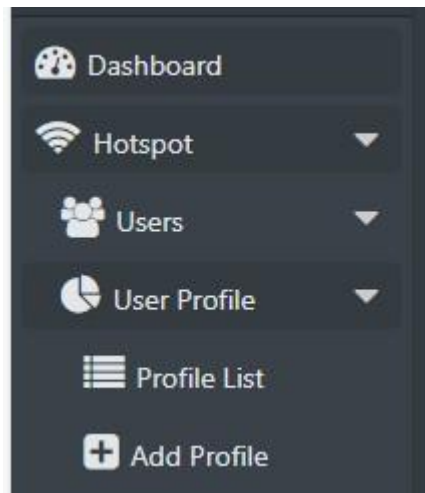
Pada menu add router isi ip router hotspot server dan sesuaikan data yang lainnya. Sesuai konfigurasi hotspot sebelumnya, lalu klik save dan connect.

### Sesuaikan dengan server hotspot

Pengoperasian Alokasi *Bandwidth* Dan Manajemen Pada Profile Tahap ini melibatkan pembuatan profil khusus untuk mengelola alokasi *bandwidth*, masa berlaku atau lease time user, deskripsi profil, serta pengaturan notifikasi jika masa penggunaan sudah habis. Profil ini memungkinkan administrator untuk mengelola dan membatasi penggunaan *bandwidth* sesuai kebutuhan, memastikan distribusi yang merata di antara pengguna. Berikut adalah tahapannya :

### Add profile pada menu dashboard

Pertama buka user manage berbasis vps pada web browser.



Add profile pada user manage

Add profile pada menu dashboard

Isi data profile yang ingin dibuat sesuai dengan gambar dibawah.

| + Add User Profile                   |                 |
|--------------------------------------|-----------------|
| <span>Close</span> <span>Save</span> |                 |
| Name                                 | KONSUMEN        |
| Address Pool                         | hs-pool-1       |
| Shared Users                         | 1               |
| Rate limit [up/down]                 | 1500k           |
| Expired Mode                         | Remove & Record |
| Validity                             | 12h             |
| Price Rp                             | 0               |
| Selling Price Rp                     | 0               |
| Lock User                            | Enable          |
| Parent Queue                         | none            |

Isi profile pada user manage

Untuk melengkapi penjelasan akan diuraikan dalam tabel dibawah yang menjelaskan fungsi setiap menu pada gambar diatas.

Profile berhasil dibuat

Satu profile sudah berhasil dibuat dengan deskripsi konsumen. Untuk cek dan mengedit profile, bisa diterapkan di menu profile list.

| 2 items | Name     | Shared Users | Rate Limit | Expired Mode    | Validity |
|---------|----------|--------------|------------|-----------------|----------|
|         | default  | 200          |            |                 |          |
|         | KONSUMEN | 1            | 1500k      | Remove & Record | 12h      |

Hasil profile



### Pengoperasian Autentikasi Dengan Generate User Dan Password

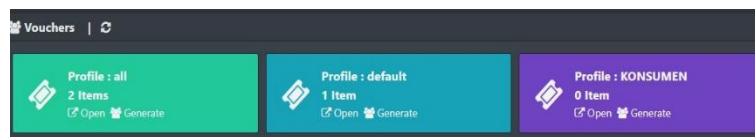
Pada tahap ini, dilakukan proses pengacakan (generate) user dan password melalui sistem sesuai profile untuk mengamankan akses ke jaringan. Autentikasi ini penting untuk memastikan hanya pengguna yang memiliki akses sah yang dapat terhubung ke jaringan, sehingga menjaga keamanan dan efisiensi jaringan.



Menu vouchers

Memilih profile yang telah dibuat

Klik generate pada list vouchers terdapat beberapa profile dengan manajemen berbeda, pilihlah profile yang ingin kita atur dan cetak autentikasinya.



Pilih Profile

Isi setingan pada generate autentikasi

Isi setingan pada generate autentikasi disesuaikan dengan gambar dibawah lalu klik Generate.

| Generate User                                      |                     |
|----------------------------------------------------|---------------------|
| Qty                                                | 20                  |
| Server                                             | hotspot1            |
| User Mode                                          | Username & Password |
| Name Length                                        | 4                   |
| Prefix                                             |                     |
| Character                                          | Random aBcD         |
| Profile                                            | KONSUMEN            |
| Time Limit                                         | 12h                 |
| Data Limit                                         | MB                  |
| Comment                                            |                     |
| Validity : 12h   Price : Rp 0   Lock User : Enable |                     |

Buat autentikasi



Setelah penerapan autentikasi, proses menghubungkan ke jaringan Wi-Fi menjadi lebih terstruktur dan aman. Karena saat menghubungkan ke jaringan, user akan diminta untuk memasukkan kode khusus di dalam form login beserta password yang telah digenerate untuk dapat mengakses jaringan Wi-Fi. Proses ini memastikan bahwa hanya pengguna yang memiliki kode dan password yang valid yang dapat terhubung.

Setelah penerapan manajemen bandwidth melalui pembuatan profil user, pengujian konektivitas diulang untuk melihat efektivitas dari settingan baru ini. Profil Wi-Fi yang telah dikelola memberikan limitasi bandwidth yang spesifik untuk setiap pengguna. Setelah penerapan manajemen bandwidth melalui pembuatan profil user, pengujian konektivitas diulang untuk melihat efektivitas dari settingan baru. Profil Wi-Fi yang telah dikelola memberikan limitasi bandwidth yang spesifik untuk setiap pengguna

## **Pembahasan**

Dalam manajemen jaringan yang baik, distribusi bandwidth yang tepat adalah hal yang penting. Secara umum, pengelolaan bandwidth sering kali diterapkan secara seragam kepada semua pengguna, namun pada sistem User Management berbasis VPS, administrator memiliki fleksibilitas untuk membuat beberapa profil bandwidth yang disesuaikan dengan kebutuhan pengguna yang berbeda. Dalam konteks optimalisasi dan efektivitas, terdapat dua kelompok utama pengguna, yaitu konsumen dan staf. Kebutuhan bandwidth untuk kedua kelompok ini sangat berbeda, di mana staf hanya membutuhkan sedikit bandwidth untuk aktivitas kantor seperti browsing, sedangkan konsumen memerlukan lebih banyak bandwidth untuk kegiatan seperti streaming video atau penggunaan media sosial.

## **PENUTUP**

### **Simpulan**

Sistem manajemen bandwidth berbasis VPS dengan integrasi API Management Services berhasil diimplementasikan menggunakan teknologi traffic control (tc) pada Linux dan framework FastAPI. Arsitektur yang dibangun terbukti mampu menyediakan kontrol bandwidth yang terpusat dan terprogram.

Sistem yang dikembangkan mampu membatasi bandwidth pengguna secara real-time dengan akurasi mencapai 95% dari nilai yang ditetapkan. Pengujian menunjukkan bahwa pembatasan kecepatan download dan upload dapat diterapkan secara efektif pada berbagai level kecepatan (mulai dari 1 Mbps hingga 100 Mbps) dengan deviasi rata-rata hanya 3-5%..

### **Saran**

Disarankan untuk menambahkan fitur monitoring dan reporting real-time yang lebih komprehensif dengan visualisasi grafik penggunaan bandwidth

Perlunya pengembangan mekanisme auto-scaling bandwidth berdasarkan waktu (time-based) dan beban jaringan

Penambahan fitur notifikasi otomatis ketika pengguna mendekati batas bandwidth yang ditetapkan.

Perlu dilakukan penelitian lebih lanjut mengenai implementasi machine learning untuk predictive bandwidth allocation



## REFERENSI

- Barovih, G., Artika, L., & Manto, F. A. (2022). Network Optimization and Network User Access Limitations Using Mikhmon Server at Diskominfo Palembang City. *LC International Journal of STEM* (ISSN: 2708-7123), 3(4), 72-82.
- Cisco Systems. (2023). Bandwidth management overview. Retrieved from <https://www.cisco.com/>
- DigitalOcean. (2022). Using API to manage VPS bandwidth and network traffic. Retrieved from <https://docs.digitalocean.com/reference/api/>
- Fadilah, H., & Ramadhan, Y. (2022). Penerapan API dalam layanan cloud untuk otomasi manajemen jaringan. *Jurnal Teknologi dan Rekayasa Sistem Komputer*, 6(3), 77–85.
- Google Cloud. (2023). Managing bandwidth and quotas using Cloud API. Retrieved from <https://cloud.google>
- Krisnaningsih, E., Dwiyatno, S., Jubaedi, A. D., & Maulana, Y. F. (2023). MANAJEMEN AKSES INTERNET BERBASIS LIMITASI MENGGUNAKAN MIKHMON DAN ROUTER MIKROTIK. *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, 10(2), 84-91.
- Kurose, J. F., & Ross, K. W. (2021). *Computer networking: A top-down approach* (8th ed.). Pearson.
- Mikrotik. (2023). RouterOS API documentation. Retrieved from <https://wiki.mikrotik.com/wiki/Manual:API>
- Nugroho, R., & Setiawan, D. (2020). Analisis implementasi virtual private server untuk layanan web kampus. *Jurnal Sistem Informasi dan Komputer*, 8(1), 12–20.
- OpenStack Foundation. (2023). API reference for network bandwidth management. Retrieved from <https://docs.openstack.org/api-ref/>
- Pratama, R. A. (2021). *Administrasi jaringan berbasis Linux dan Mikrotik*. Yogyakarta: Deepublish.
- Santoso, D., & Prasetyo, A. (2022). Implementasi manajemen bandwidth menggunakan Mikrotik dan API. *Jurnal Teknologi Informasi dan Komputer*, 9(2), 110–118.
- Sari, M. N., & Rachman, A. (2021). Pengelolaan bandwidth berbasis cloud untuk efisiensi jaringan kampus. *Jurnal Ilmiah Teknologi dan Sistem Informasi*, 7(1), 45–53.
- Stallings, W. (2013). *Foundations of modern networking: SDN, NFV, QoE, IoT, and cloud*. Addison-Wesley.
- Tanenbaum, A. S., & Wetherall, D. J. (2011). *Computer networks* (5th ed.). Pearson Education.
- Ubuntu. (2022). Network bandwidth management in Linux servers. Canonical Documentation. Retrieved from <https://ubuntu.com/server/docs>
- Wahyudi, A. (2020). *Jaringan komputer: Konsep dasar dan implementasi*. Jakarta: Mitra Wacana Media.