

OPTIMASI RADIUS SERVER UNTUK PENGATURAN ALOKASI BANDWIDTH PADA JARINGAN HOTSPOT

I Putu Agus Eka Pratama, I Gede Bagus Premana Putra*

Fakultas Teknik, Universitas Udayana
email: baguspramana17@gmail.com

Abstract

Hotspot is one form of utilizing Wireless LAN technology that can be used to access internet services and is usually found in public areas such as libraries, campus internet parks, or offices. Aspects that need to be considered in hotspots to make users feel comfortable is security then at the provider side, the aspect that needs to be considered is the regulation of bandwidth allocation to optimize the data transfer speeds that the network and to prevent the possibility of dense network traffic. RADIUS server is one type of server that can be used at hotspots to secure hotspots because it supports various types of encryption. In this study, optimization of the settings for hotspot network bandwidth is optimized by integrating the RADIUS server with RouterOS. Bandwidth allocation management is done by determining the active time of a user account and set quotas hotspot uploads and downloads for the account. The results obtained from this study indicate that when a hotspot user account has passed the active time or has passed the upload quota and the download given, the account will be deleted from the list of hotspot user accounts or disabled.

Keywords: hotspot, RADIUS server, bandwidth allocation settings, routerOS

1. PENDAHULUAN

Komunikasi nirkabel atau tanpa kabel (*wireless*) merupakan salah satu kebutuhan yang tidak bisa dilepaskan dari kehidupan masyarakat pada era sekarang. Komunikasi tanpa kabel yang berkembang dimasyarakat memiliki banyak jenis salah satunya adalah LAN nirkabel atau yang lebih dikenal dengan sebutan Wi-Fi. Wi-Fi menjadi teknologi alternatif dan sangat mudah untuk diimplementasi diberbagai lokasi, seperti perkantoran, sekolah, laboratorium, atau di rumah sekalipun. Wi-Fi juga berkaitan erat dengan suatu area yang dapat dimanfaatkan untuk mengakses layanan Internet, sehingga tidak jarang Wi-Fi juga disebut dengan *hotspot*, terutama yang tersedia di wilayah *public*. Penggunaan jaringan Wi-Fi/*hotspot* juga lebih fleksibel karena perangkat yang ingin terhubung ke jaringan Wi-Fi/*hotspot* tidak perlu menggunakan kabel. Melihat banyaknya kemudahan yang ditawarkan oleh Wi-Fi/*hotspot*, menyebabkan banyak orang yang tertarik untuk menggunakannya (Priyamdo 2005).

Jaringan Wi-Fi/*hotspot* selain menawarkan banyak kemudahan, teknologi ini juga memiliki kelemahan. Salah satu kelemahan atau dapat dikatakan sebagai suatu masalah yang akan dihadapi apabila

menerapkan Wi-Fi adalah isu keamanan. Melihat hal tersebut, maka sampai sekarang masih banyak pihak yang mempertanyakan tentang sistem keamanan dari Wi-Fi, terutama dalam implementasinya. Apabila sistem keamanan ini diabaikan, maka terdapat kemungkinan adalah pengguna yang tidak berhak (*illegal*) dapat masuk atau menggunakan layanan yang dimiliki oleh *hotspot* (A. Ahmad 2015).

Manajemen pengguna juga merupakan permasalahan yang tidak bisa dilepaskan dari jaringan *hotspot*. Permasalahan ini lahir, karena jaringan *hotspot* sangat digemari oleh masyarakat dan mudah untuk digunakan, sehingga dapat dikatakan manajemen pengguna ini sebagai tantangan bagi administrator dalam implementasi *hotspot*. Manajemen pengguna yang dimaksud adalah yang pertama manajemen terhadap pengguna yang berhak mengakses layanan yang dimiliki jaringan (dapat diterapkan dengan sistem akun) dan manajemen kedua adalah pengaturan *bandwidth*, dengan tujuan untuk menghindari adanya kemungkinan monopoli *bandwidth* oleh *user* tertentu (M. Solikhul 2016).

Masalah keamanan yang dimiliki oleh jaringan *hotspot* dapat diatasi dengan mengintegrasikan atau menerapkan RADIUS

(*Remote Access Dial In User Service*) server. RADIUS pertama kali dikembangkan oleh Livingston Enterprise, yang selanjutnya disebut sebagai protokol keamanan jaringan dan dapat digunakan untuk manajemen akses Internet bagi *client*. RADIUS melakukan manajemen akses *client* dengan tiga metode yaitu otentikasi, otorisasi, dan pendaftaran akun pengguna secara terpusat. RADIUS menggunakan standar IEEE 802.1x atau sering disebut "*Port Based Authentication*" dan RADIUS beroperasi di layer aplikasi model referensi OSI (Hakan 2002).

RouterOS merupakan produk yang dikembangkan oleh Mikrotik yang diperuntukan sebagai *network router*. RouterOS dibuat dengan kernel Linux, sehingga membuat setiap perintah yang terdapat berbasis *command line*, RouterOS juga dapat diinstal pada PC yang menyebabkan PC tersebut dapat difungsikan sebagai *router* jaringan yang handal. RouterOS didesain untuk memberikan kemudahan bagi pengguna dan telah memiliki *interfaces* berbasis GUI melalui aplikasi WinBox. Fitur yang dimiliki oleh RouterOS sangat lengkap, terutama untuk membangun sebuah jaringan yang handal, adapun fitur tersebut diantaranya sebagai *gateway* Internet, *routing*, *firewall*, *hotspot*, sampai pengaturan alokasi *bandwidth* (Khoirul 2017).

Penelitian yang dilakukan oleh (A. Cristescu 2016) yaitu implementasi RADIUS server sebagai solusi untuk proses autentikasi yang legal. Pada penelitian ini dilakukan realisasi dari solusi mengenai *authentication*, *authorization* dan *accounting user* untuk mengakses internet pada sebuah jaringan *wireless* dengan memanfaatkan RADIUS server. RADIUS server yang diterapkan akan melakukan enkapsulasi paket dengan metode PAP, CHAP, MSCHAPv1 dan MSCHAPv2 untuk menciptakan suatu protokol atau algoritma yang mampu melindungi data *user*.

Penelitian selanjutnya oleh (A. Rahman 2016) yaitu implementasi manajemen *bandwidth* berbasis autentikasi *user*. Pada penelitian ini dilakukan implementasi RADIUS server yang diintegrasikan dengan Mikrotik untuk keperluan manajemen *bandwidth* yang mengambil studi kasus di Universitas Mulawarman. Cara implementasi yang dilakukan pada penelitian ini adalah

dengan eksperimen untuk menciptakan suatu sistem yang mampu mengontrol dan mengalokasikan *bandwidth* kepada *user* berdasarkan jabatan (dosen, pegawai dan mahasiswa) saat mengakses jaringan Wi-Fi dan diintegrasikan dengan RADIUS server untuk keperluan autentikasi *user* yang legal.

Penelitian ini akan dilakukan optimasi pengaturan alokasi *bandwidth* dengan cara mengintegrasikan RADIUS server dengan Mikrotik RouterOS. Cara yang digunakan untuk pengaturan alokasi *bandwidth* adalah pertama menentukan masa aktif dari suatu akun pengguna jaringan *hotspot* dan kedua menentukan kuota *upload* serta *download* dari suatu akun pengguna jaringan *hotspot*, jika akun yang menggunakan atau terhubung dengan jaringan *hotspot* telah melewati batas waktu atau telah melewati batas kuota *upload* dan *download* yang telah ditentukan, maka akun tersebut akan di-*disable* dan hanya dapat digunakan kembali setelah dua jam dari waktu *disable*.

2. METODE PENELITIAN

Pembuatan sistem optimasi RADIUS server untuk manajemen bandwidth pada jaringan *hotspot* memerlukan beberapa metode penelitian, adapun metode yang digunakan pada penelitian ini adalah

1. Mendeskripsikan masalah utama dan menetapkan tujuan pokok dalam penelitian, yang sebelumnya sudah dijelaskan pada bagian pendahuluan.
2. Mengumpulkan data dan literatur yang diperlukan untuk memperkuat penelitian, yang dilakukan dengan beberapa metode berikut
 - a. Wawancara, wawancara dilakukan dengan pengguna layanan *hotspot* terkait tingkat kenyamanan layanan yang diterimanya, serta pihak pengelola layanan, untuk mengetahui cara yang diterapkan untuk melakukan manajemen *user*.
 - b. Studi pustaka, pencarian dan pengumpulan data yang diperlukan untuk memperkuat penelitian ini, baik berasal dari buku ataupun jurnal *online*.

3. Pengembangan sistem optimasi RADIUS *server* yang terbagi menjadi 4 tahap, yaitu
 - a. Analisis kebutuhan sistem, dengan tujuan untuk mendefinisikan ruang lingkup dan perangkat keras serta lunak yang diperlukan dalam pengembangan sistem.
 - b. Desain, untuk menentukan desain topologi jaringan yang akan digunakan serta mendesain fungsi dari setiap *interfaces* yang dimiliki oleh perangkat keras.
 - c. Implementasi, merupakan tahap penerapan hasil analisis dan desain menjadi sistem optimasi RADIUS *server*, yang mana implementasi dilakukan dengan memanfaatkan virtualisasi.
 - d. *Testing*, untuk melakukan validasi dan verifikasi terhadap implementasi yang telah dilakukan.

3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan yang akan dilakukan pada penelitian ini diawali dengan menganalisis kebutuhan sistem, membuat topologi sistem, menentukan fungsi dari setiap *interfaces* yang dimiliki oleh RouterOS, melakukan implementasi topologi sistem yang telah dibuat dan terakhir melakukan uji coba sistem.

3.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan dengan cara menentukan perangkat keras dan perangkat lunak yang diperlukan oleh sistem optimasi RADIUS *server*, yang dapat dijelaskan sebagai berikut.

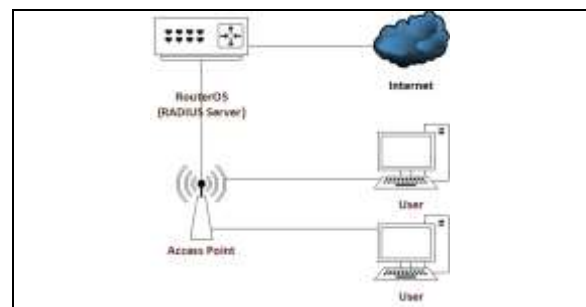
Tabel 1 Kebutuhan Sistem

No	Perangkat	Fungsi
1	Sumber Internet	Sebagai sumber koneksi internet yang akan digunakan oleh <i>client</i> .
2	RouterOS	Pusat dari sistem, yang memiliki fitur untuk melakukan manajemen <i>user</i> dan <i>bandwidth</i> berbasis

No	Perangkat	Fungsi
		RADIUS <i>server</i> .
3	Access Point	Sebagai pemancar jaringan <i>hotspot</i> .
4	Client	Diperlukan saat tahap <i>testing</i> , untuk verifikasi dan validasi sistem optimasi RADIUS <i>server</i> pada penelitian ini

3.2 Topologi Sistem

Model topologi yang digunakan pada penelitian ini adalah topologi *star*. Mikrotik RouterOS menjadi pusat dari jaringan *hotspot* yang menghubungkan *user* dengan Internet. Mikrotik RouterOS yang digunakan juga telah dilengkapi dengan fungsi limitasi waktu penggunaan dan limitasi kuota *upload* serta *download* bagi *user hotspot*. Pengujian dilakukan terhadap fungsi limitasi waktu penggunaan (*uptime*), kuota *upload* (*byte-in*) dan kuota *download* (*byte-out*).



Gambar 1 Gambaran Umum Sistem

Gambar 1 menunjukkan topologi sistem dari penelitian yang akan digunakan. Topologi yang digunakan adalah *star* dengan RouterOS sebagai pusat jaringan *hotspot* yang menghubungkan *client* ke jaringan Internet dengan bantuan *access point* sebagai pemancar jaringan *hotspot*.

3.3 Fungsi Setiap Interfaces RouterOS

RouterOS yang digunakan pada penelitian ini telah diinstal pada PC dengan menerapkan metode virtualisasi menggunakan *software* VirtualBox. RouterOS yang telah diinstal memiliki tiga *interfaces* yang digunakan untuk keperluan jaringan *hotspot*, adapun

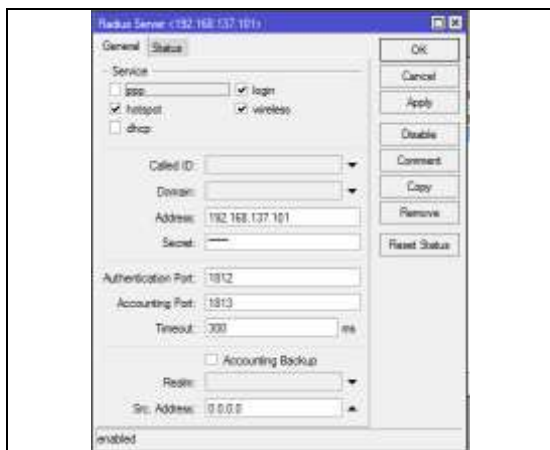
detail dari *interfaces* yang dimiliki oleh RouterOS adalah seperti berikut.

Tabel 2 Fungsi Interfaces RouterOS

No	Nama Interfaces	Mode	Fungsi
1	Ether-1	NAT	Menghubungkan RouterOS dengan koneksi internet yang dimiliki oleh PC
2	Ether-2	Ter-Bridge Adapter	Menghubungkan RouterOS dengan Access Point (AP berfungsi sebagai pemancar hotspot)
3	Ether-3	Host Only Adapter	Menghubungkan RouterOS dengan PC yang digunakan untuk keperluan konfigurasi

3.4 Implementasi

Implementasi dari jaringan *hotspot* dilakukan dengan menerapkan virtualisasi RouterOS pada PC sebagai *router* yang berfungsi sebagai *controller* jaringan *hotspot*. Jaringan *hotspot* yang telah dibuat juga telah diintegrasikan dengan RADIUS server yang dimiliki oleh RouterOS.



Gambar 2 Penggunaan RADIUS Server pada Jaringan Hotspot

Gambar 2 menunjukkan cara penggunaan atau integrasi jaringan *hotspot* dengan RADIUS server yang dimiliki oleh RouterOS.



Gambar 3 Script Limitasi Batas Waktu Penggunaan Hotspot (Uptime)

Gambar 3 menunjukkan *script* yang digunakan untuk melakukan limitasi batas waktu penggunaan jaringan *hotspot (uptime)* bagi suatu *user*. *Script* untuk limitasi batas waktu penggunaan jaringan *hotspot* akan dijalankan secara otomatis ketika *user* telah berhasil melakukan *login*, maka dari itu *script* tersebut ditempatkan pada bagian *on-login*. *Script* limitasi *uptime* yang terdapat pada Gambar 3 akan menghapus *user* yang telah *login* ketika *uptime*-nya telah mencapai satu menit, kemudian *user* tersebut akan dihapus pada hari berikutnya untuk menghindari kemungkinan penyimpanan yang penuh dengan *user* yang sudah tidak aktif.



Gambar 4 Script Limitasi Kuota Upload (byte-in)

Gambar 4 menunjukkan *script* yang digunakan untuk melakukan limitasi kuota

upload (byte-in) user hotspot ketika berhasil melakukan *login*. *Script* yang terdapat pada Gambar 4 akan menghapus *user* yang sedang aktif ketika kouta *upload*-nya telah mencapai 10 MB, kemudian *user* tersebut akan dihapus pada hari berikutnya.



Gambar 5 *Script* Limitasi Kuota Download (*byte-out*)

Gambar 5 menunjukkan *script* yang digunakan untuk melakukan limitasi kuota *download (byte-out) user hotspot* ketika berhasil melakukan *login*. *Script* yang terdapat pada Gambar 5 akan menghapus *user* yang sedang aktif ketika kouta *download*-nya telah mencapai 10 MB, kemudian *user* tersebut akan dihapus pada hari berikutnya.



Gambar 6 *Script* *User Trial*

Gambar 6 menunjukkan *script* yang digunakan untuk menciptakan akun *user* yang bersifat *trial*. *Script user trial* akan membatasi waktu penggunaan layanan jaringan selama satu menit, setelah satu menit dari waktu *login user* tersebut akan dihapus dari daftar *user* aktif dan di-setting menjadi *disable*. Akun *user* akan aktif kembali pada hari berikut.

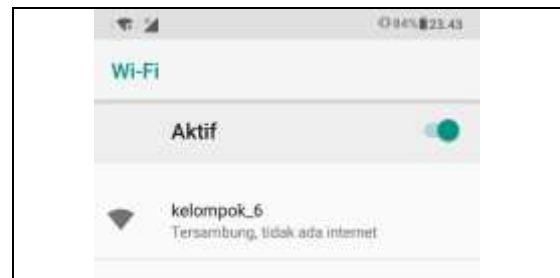
3.5 Uji Coba

Langkah selanjutnya setelah melakukan implementasi jaringan *hotspot* adalah uji coba jaringan *hotspot* yang telah dibuat pada sisi *user*. Akun *user* yang dapat digunakan untuk mengakses layanan jaringan *hotspot* adalah sebagai berikut.



Gambar 7 Akun *User*

Gambar 7 menunjukkan akun yang dapat dipakai oleh *user* untuk mengakses jaringan *hotspot*. Akun *user* yang dimiliki oleh RouterOS berjumlah empat dengan *profile* yang berbeda sesuai dengan empat *user profile* yang dimiliki oleh RouterOS.



Gambar 8 Koneksi *User* ke *Hotspot*

Gambar 8 menunjukkan pengaksesan jaringan *hotspot* oleh *user*. Jaringan *hotspot* yang telah dibuat memiliki SSID kelompok_6.



Gambar 9 Pesan Gagal *Login*

Gambar 9 menunjukkan pesan yang ditampilkan ketika *user* gagal melakukan *login* ke jaringan *hotspot*. Pesan yang ditampilkan pada Gambar 9 menunjukkan bahwa penerapan RADIUS *server* pada jaringan *hotspot* telah berhasil.



Gambar 10 Login User Trial

Gambar 10 menunjukkan *user login* dengan jaringan *hotspot* dengan akun yang bersifat *trial*. Akun yang bersifat *trial* akan memberikan *user* untuk menggunakan jaringan *hotspot* selama satu menit, setelah satu menit akun *user* akan di-*disable* dan di-*enable* kembali pada hari berikutnya.



Gambar 11 Hasil Trial User

Gambar 11 menunjukkan akun *trial user* yang di-*disable* setelah satu menit. Akun tersebut akan di-*enable* kembali pada hari berikut.



Gambar 12 Login User dengan Limitasi Uptime

Gambar 12 menunjukkan akun *user* yang berhasil *login* dengan limitasi *uptime*. Limitasi *uptime* akan menampilkan batas waktu penggunaan yang tersisa bagi *user*

untuk menggunakan jaringan *hotspot*. Akun *user* yang telah mencapai batas limitasi *uptime* akan dihapus pada hari berikut.



Gambar 13 Pesan Error Akun yang Telah Mencapai Uptime

Gambar 13 menunjukkan pesan yang ditampilkan ketika *user* telah mencapai batas waktu *uptime* yang telah ditentukan. Akun *user* yang telah mencapai batas *uptime* tidak bisa untuk *login* kembali dan akan ditampilkan pesan *error* berupa peringatan bahwa akun tersebut telah mencapai batas *uptime* dan dinyatakan sudah tidak aktif.

4. KESIMPULAN

Berdasarkan hasil dan pembahasan diatas, maka penulis dapat mengambil kesimpulan sebagai berikut.

1. Optimasi RADIUS *server* untuk pengaturan alokasi *bandwidth* merupakan sistem yang dibuat guna membantu administrator jaringan dalam melakukan pengamanan dan manajemen *user* terhadap pengguna layanan *hotspot*.
2. *Security* tambahan yang dapat diberikan pada jaringan *hotspot* untuk menangani autentikasi dan otorisasi *user* adalah RADIUS *server*.
3. *Bandwidth* manajemen *user hotspot* dapat dilakukan dengan memanfaatkan fitur *script* pada *user profile* Mikrotik *hotspot*.
4. Pengujian fungsi limitasi batas waktu penggunaan (*uptime*) layanan jaringan *hotspot* oleh *user* dapat dilakukan

dengan memanfaatkan fitur *script on-login* yang dimiliki oleh *user profile* RouterOS, yang mana pada bagian *script* ini dapat ditentukan *limit-uptime* yang ingin diberikan atau menggunakan sistem interval yang dihitung setelah *user* melakukan *login*.

5. Fungsi limitasi batas kuota *upload* dan batas kuota *download* juga dapat diterapkan dengan menggunakan fitur *script on-login* yang dimiliki oleh RouterOS, yang mana pada bagian *script* dapat ditentukan *limit-byte-out* untuk limitasi kuota *download* dan *limit-byte-in* untuk limitasi kuota *upload*.

5. REFERENSI

- A. Ahmad, Kusnawi. 2015. *Analisa Keamanan Jaringan Wireless Menggunakan Radius Server pada Mikrotik (Studi Kasus : Perpustakaan Universitas Gadjah Mada)*. Yogyakarta.
- A. Cristescu, V. Sorici. 2016. "Implementing an AAA-RADIUS Solution Based on Legacy Authentication Protocols."
- A. Rahman, Haviluddin. 2016. "Implementation of Bandwidth Management Authentication." *Internation Journal of Computing and Informatics (IJCANDI)* 1-8.
- Hakan, V. 2002. "DIAMETERNext Generation's AAA Protocol." *www.diva-portal.org*. Diakses November 2019, 16. <http://www.diva-portal.org/smash/get/diva2:18347/FULLTEXT01.pdf>.
- Khoirul, A. 2017. *Rancang Bangun Sistem Manajemen User Hotspot Menggunakan Mikrotik PHP API Berbasis Web di Pondok Pesantren Al-Luqmaniyyah*. Yogyakarta: Universitas PGRI Yogyakarta.
- M. Solikhul, T. I. Bayu. 2016. *Perancangan dan Implementasi Manajemen User pada Jaringan Hotspot Mikrotik (Studi Kasus Kantor Perpustakaan dan Arsip Daerah Kota Salatiga)*. Salatiga: Universitas Kristen Satya Wacana.
- Priyamdo, T.K dan Heriadi, D. 2005. *Jaringan Wi-Fi Teori dan Implementasi*. Yogyakarta: Andi.