Implementasi Server RADIUS dan Metode Per Connection Classifier (PCC) untuk Autentikasi dan Load Balancing pada Jaringan Enterprise

1st Rafael Christian Marbun
Teknologi Informasi
Telkom University Surabaya
Surabaya, Indonesia
rafaelchrist@student.telkomuniversity.ac.id

2nd Muhammad Adib Kamali
Teknologi Informasi
Telkom University Surabaya
Surabaya, Indonesia
adibmkamali@telkomuniversity.ac.id

3rd Philip Tobianto Daely
Teknologi Informasi
Telkom University Surabaya
Surabaya, Indonesia
ptdaely@telkomuniversity.ac.id

Abstrak **— Penelitian ini merancang dan mengimplementasikan sebuah solusi jaringan untuk lingkungan *enterprise* guna mengatasi dua masalah utama yaitu, kerentanan keamanan yang timbul dari penggunaan kata sandi bersama dan ketidakstabilan koneksi internet. Sistem yang dikembangkan memanfaatkan perangkat MikroTik sebagai router dan Raspberry Pi sebagai server, dengan tujuan menciptakan jaringan yang aman, andal, dan efisien. Untuk meningkatkan keamanan, sistem ini mengganti metode autentikasi berbasis kata sandi tunggal dengan sistem autentikasi terpusat per-pengguna melalui Server RADIUS yang dipadukan dengan protokol keamanan WPA2 Enterprise. Hasil pengujian membuktikan bahwa pendekatan ini efektif melindungi kredensial pengguna dari serangan *evil twin*. Selain itu, proses autentikasi terbukti sangat cepat, dengan kecepatan rata-rata tercatat hanya 0,8 ms per pengguna. Dari sisi performa jaringan, metode *Load Balancing* Per Connection Classifier (PCC) berhasil diimplementasikan untuk mengoptimalkan dan menggabungkan dua jalur koneksi ISP. Hal ini menghasilkan peningkatan *throughput* gabungan yang signifikan, mencapai 66,29 Mbps untuk aktivitas unduh dan 49,01 Mbps untuk aktivitas unggah. Keandalan sistem dijamin melalui mekanisme *failover* ganda. Pengujian menunjukkan bahwa sistem mampu melakukan *failover* koneksi jaringan (jika salah satu ISP putus) dengan waktu henti (*downtime*) rata-rata hanya 3 detik. Sementara itu, untuk kegagalan pada server autentikasi, sistem dapat beralih ke Server RADIUS cadangan dalam waktu 5 detik. Secara keseluruhan, hasil penelitian membuktikan bahwa solusi yang diimplementasikan ini efektif untuk diterapkan sebagai sistem jaringan *enterprise* yang aman, stabil, dan memiliki kinerja yang andal.**

Kata kunci**— Jaringan Enterprise, Keamanan jaringan, Server RADIUS, Load Balancing, Per Connection Classifier (PCC), Autentikasi, WPA2 Enterprise**

I. PENDAHULUAN

Di era kompleksitas digital yang semakin meningkat, kebutuhan akan sistem keamanan jaringan yang andal telah menjadi lebih krusial dari sebelumnya. Banyak perusahaan dan organisasi, baik skala kecil maupun besar, menerapkan fitur autentikasi dasar pada sistem jaringan mereka seperti menggunakan kata sandi yang sama untuk autentikasi jaringan. Mereka tidak memiliki langkah-langkah autentikasi yang ketat dan hanya menerapkan pemblokiran berdasarkan alamat MAC. Sebuah studi tahun 2020 oleh Positive Technology Software Company, sebuah firma riset keamanan di Moskow, Rusia, mengungkapkan bahwa 88% jaringan nirkabel lokal di sektor industri, pemerintahan, dan telekomunikasi memiliki kerentanan yang sangat serius. Dari kerentanan ini, 63% terkait dengan kelemahan dalam kebijakan kata sandi, 75% melibatkan penggunaan kata sandi yang sudah diketahui, dan 25% disebabkan oleh kurangnya autentikasi pada jaringan-jaringan tersebut [1].

Pada tahun 2023, data Bank Dunia menunjukkan bahwa 69% dari total populasi Indonesia merupakan pengguna internet aktif [2]. Pertumbuhan pesat ini dapat berdampak pada beban trafik karena server menjadi kelebihan beban. Di sisi lain, perusahaan, institusi pendidikan, dan kafe sering kali memanfaatkan lebih dari satu jaringan untuk memenuhi kebutuhan mereka. Namun, masalah muncul ketika salah satu dari jaringan tersebut mati atau terputus. Untuk mengatasi masalah ini, menerapkan *load balancing* merupakan solusi yang tepat. Metode ini dapat beroperasi secara efektif jika permintaan dari klien dapat dikelola secara merata, sehingga memungkinkan trafik mengalir secara optimal, meningkatkan kinerja dan mengurangi waktu respons untuk mencegah beban berlebih pada satu jalur koneksi tunggal [3].

Sistem jaringan yang dirancang memiliki keamanan yang tinggi dengan penggunaan *Remote Access Dial-In User Service* (RADIUS), yang merupakan protokol dengan fitur *Authentication, Authorization, Accounting* (AAA), berdasarkan mekanisme autentikasi metode yang sering digunakan yaitu *Challenge/Response*, yang dapat memastikan bahwa pengguna yang mencoba mengakses sistem memang memiliki hak akses masuk [4]. Dengan tambahan fitur *failover* pada server RADIUS sebagai cadangan jika server utama terjadi permasalahan. Dan metode *load balancing Per Connection Classifier (*PCC*)* untuk mengatasi kelebihan beban jaringan [5].

Penelitian ini bertujuan untuk mengimplementasikan jaringan terpusat dan pembagi beban jaringan. Yang diharapkan dapat memberikan kontrol akses yang lebih ketat terhadap pengguna jaringan, sehingga meminimalkan risiko

pelanggaran keamanan. Implementasi *failover* pada server RADIUS juga diharapkan dapat meningkatkan keandalan sistem, memungkinkan pemulihan cepat dari gangguan dan menjaga kontinuitas layanan. Beroperasi secara optimal dengan distribusi beban yang merata, sehingga mengurangi kemungkinan terjadinya *bottleneck* dan meningkatkan pengalaman pengguna secara keseluruhan.

II. KAJIAN TEORI

Berikut ini merupakan beberapa studi sebelumnya telah mengeksplorasi implementasi server RADIUS dalam meningkatkan keamanan jaringan, dengan fokus pada autentikasi dan perlindungan terhadap serangan siber. Penelitian dari Indah & Wardana pada tahun 2020, mengimplementasikan server RADIUS untuk mengamankan WiFi di Jurusan Teknik Elektro Politeknik Negeri Bali. Hasil pengujian Quality of Service (QoS) menunjukkan performa jaringan yang baik dengan rincian sebagai berikut: sinyal AP 53 dBm, bandwidth 1036 bytes, throughput 198.735 bytes, delay 14 ms, jitter 36,777 ms, dan packet loss sebesar 3 persen [6].

Penelitian selanjutnya dari Kovtsur pada tahun 2022 membahas penggunaan server RADIUS untuk mengamankan layanan IPTV melalui autentikasi dan otorisasi pengguna. Sistem ini memastikan hanya pengguna terdaftar yang dapat mengakses layanan tersebut Hasil penelitian menunjukkan bahwa waktu akses layanan sangat dipengaruhi oleh beban respons yang tinggi pada server RADIUS. Oleh karena itu, disarankan untuk mengurangi beban dengan cara menyimpan respons RADIUS di *switch* [7].

Penelitian oleh Yudhistira & Harwahyu pada 2024 mengkaji strategi keamanan jaringan di sebuah perusahaan *fintech* dengan mengimplementasikan server daloRADIUS untuk memperkuat otorisasi dan Pi-hole DNS Server untuk memblokir akses ke situs-situs ilegal seperti pornografi dan perjudian online. Hasilnya menunjukkan bahwa kombinasi ini berhasil mencegah pengguna mengakses situs terlarang, sementara penelitian juga menganalisis penggunaan sumber daya server untuk 10 hingga 300 pengguna aktif [8].

Selanjutnya penelitian dari Saputra pada tahun 2021 membahas implementasi server RADIUS berbasis *cloud computing* yang dikelola dengan router MikroTik untuk mengatasi masalah antrean pengguna internet di sebuah institusi. Hasil penelitian menunjukkan bahwa solusi *cloud* ini mempermudah pengelolaan pengguna *hotspot*, menjamin distribusi *bandwidth* yang konsisten, dan memiliki akses *query database* yang lebih cepat hingga 75,4 ms dibandingkan server non-*cloud*. Meskipun demikian, penelitian ini juga mengidentifikasi kelemahan utama, yaitu ketergantungan penuh pada koneksi internet untuk proses autentikasi, yang akan gagal jika terjadi gangguan. Selain itu, waktu autentikasinya tercatat sedikit lebih lambat sekitar 2 ms dibandingkan server lokal [9].

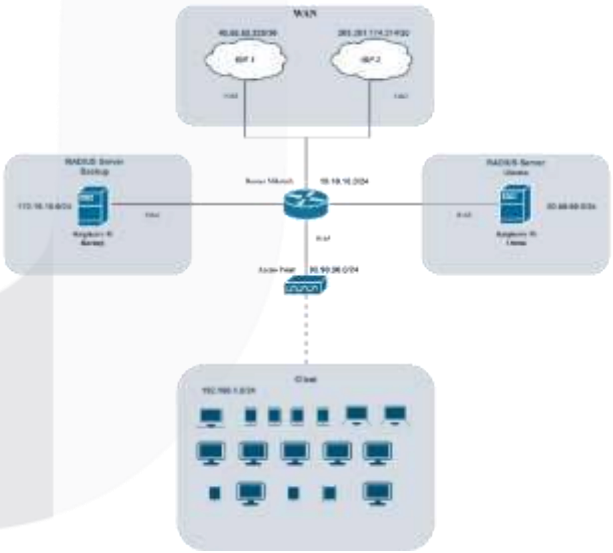
Dan yang terakhir adalah penelitian oleh Sumiah pada tahun 2023 membahas penerapan sistem autentikasi *captive portal hotspot* yang memanfaatkan server RADIUS untuk penyimpanan kredensial pengguna secara terpusat. Untuk menghubungkan beberapa lokasi yang berbeda secara aman, penelitian ini mengkaji penggunaan teknologi VPN L2TP agar dapat mengakses *database* RADIUS yang berada di pusat data melalui IP publik. Hasilnya menunjukkan bahwa setiap lokasi yang terhubung melalui VPN berhasil berkomunikasi dengan *database* pusat dan melakukan autentikasi pengguna melalui *hotspot login* pada router MikroTik, yang membuktikan keberhasilan pengelolaan data terpusat [10].

Berdasarkan penelitian tersebut, Perbedaan implementasi yang dilakukan terletak pada penggunaan RADIUS untuk koneksi ke jaringan. Penelitian sebelumnya untuk mengakses internet menggunakan *hotspot captive portal* sehingga pengguna ketika akan mengakses *Service Set Identifier* (SSID) akan diarahkan menuju *portal* untuk akses jaringan. Sedangkan yang akan dilakukan menggunakan WPA2 *Enterprise* sehingga *user* akan langsung memasukkan *username* dan *password* ketika *user* mengakses SSID tanpa harus menuju portal, kemudian dengan penggunaan dua ISP yang terkonfigurasi *load balancing* dapat dimanfaatkan sebagai pembagi beban jaringan.

III. METODE

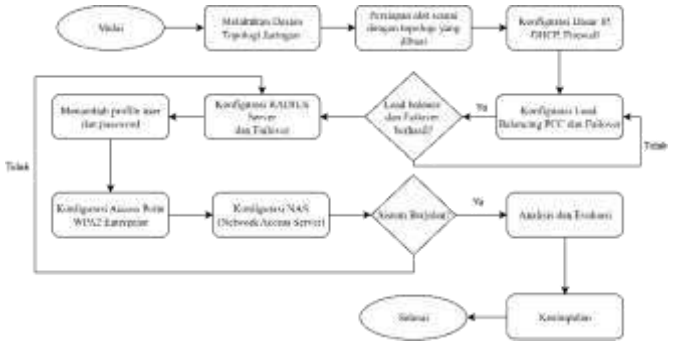
A. Topologi Jaringan

Pada gambar 1 Topologi jaringan ini berpusat pada sebuah router MikroTik yang terhubung ke dua ISP berbeda untuk menerapkan *load balancing* PCC dan *failover*. Sistem autentikasi ditangani oleh dua perangkat Raspberry Pi yang berfungsi sebagai server RADIUS utama dan cadangan. Klien terhubung ke jaringan melalui sebuah *Access Point* (AP) yang diamankan dengan protokol WPA2 Enterprise, di mana setiap permintaan autentikasi divalidasi oleh server RADIUS. Seluruh konfigurasi, termasuk komunikasi antara router dengan server RADIUS, diatur melalui antarmuka MikroTik.



GAMBAR 1
(TOPOLOGI JARINGAN)

B. Alur Perancangan Sistem



GAMBAR 2
(ALUR PERANCANGAN SISTEM)

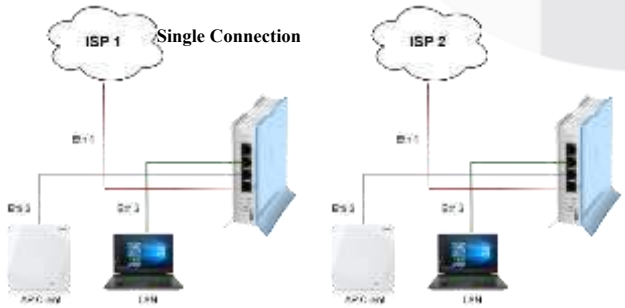
Pada gambar 2 menunjukkan Proses perancangan sistem diawali dengan tahap fundamental yaitu mendesain topologi jaringan, yang dilanjutkan dengan persiapan seluruh perangkat keras sesuai desain tersebut. Setelah itu, dilakukan konfigurasi dasar pada router MikroTik yang mencakup pengaturan IP Address, DHCP, dan *firewall*. Langkah selanjutnya adalah mengkonfigurasi sistem *load balancing* menggunakan metode PCC dan mekanisme *failover*, yang kemudian diuji untuk memastikan fungsionalitasnya. sistem autentikasi dibangun dengan mengkonfigurasi dua server RADIUS, satu sebagai server utama dan satu lagi sebagai *backup* untuk *failover* termasuk pengaturan *database* dan penjadwalan *restart* otomatis menggunakan *crontab*. Sebagai tahap akhir, profil pengguna beserta kata sandi ditambahkan ke dalam server RADIUS untuk melengkapi sistem.

C. Pengujian Sistem

Pada bagian ini sistem dilakukan pengujian dalam tiga alur yang berbeda untuk memastikan kinerja, keandalan dan keamanan sistem sebagai berikut:

1. Pengujian Kinerja Load Balancing dan Failover Jaringan

Pengujian ini menggunakan dua koneksi ISP Indihome dengan jalur yang berbeda. Tujuan dari *Load Balancing* PCC ini adalah untuk membandingkan pembagian beban dalam dua skenario yang berbeda yaitu *single connection* seperti gambar 3 dan *multi connection* gambar 4, dengan mengukur nilai *throughput* dari aktivitas *download* dan *upload* pada setiap skenario. Untuk pengujian ini sendiri akan menggunakan website dari *nperf.com* yang merupakan website untuk melakukan pengujian jaringan.



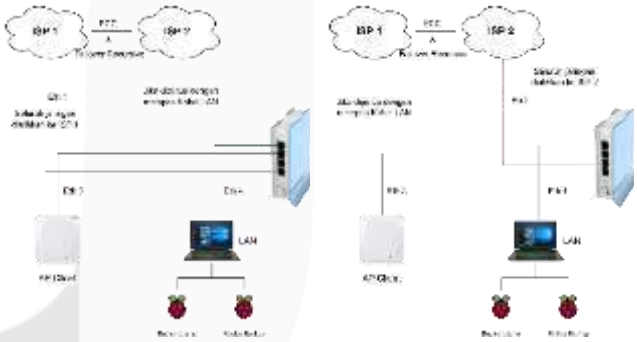
GAMBAR 3
(PENGUJIAN SINGLE CONNECTION)

GAMBAR 4
(PENGUJIAN MULTI CONNECTION)

Pengujian *failover* pada penelitian ini menerapkan metode *failover recursive*. Metode ini bekerja ketika sebuah *router* tidak mampu terhubung ke suatu *hop* DNS publik, misalnya milik Google, maka sistem akan secara otomatis mencari *hop* lain yang dapat dijangkau pada DNS publik berbeda, seperti Cloudflare. Proses pengujian dilakukan dengan memanfaatkan fitur Netwatch dari MikroTik yang berfungsi untuk melakukan *monitoring* terhadap *host* jaringan. Parameter utama yang dicatat adalah *downtime*, yaitu durasi waktu yang dibutuhkan sistem untuk beralih ke koneksi ISP yang masih aktif. Pengukuran ini dilakukan sebanyak lima kali untuk kemudian diambil nilai rata-ratanya.

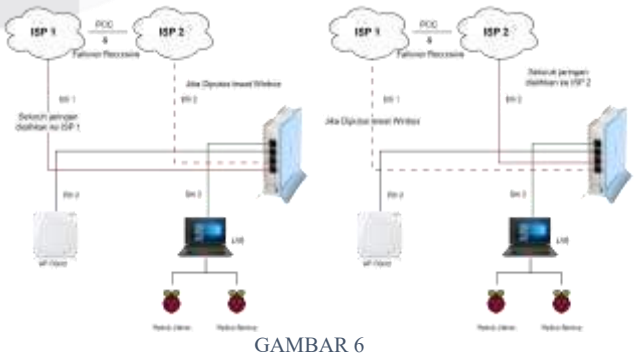
Untuk simulasi failover akan dilakukan dirouter MikroTik dengan skenario berikut:

a) Skenario pertama, ketika kabel LAN salah satu ISP dicabut, apakah nantinya ISP yang masih terhubung dengan jaringan bisa melakukan *backup* atau *failover* seperti pada gambar 5.



GAMBAR 5
(SKENARIO PERTAMA PENGUJIAN *FAILOVER* JARINGAN)

b) Skenario kedua, Ketika salah satu *interface* jaringan diputus menggunakan aplikasi winbox dari router apakah jaringan yang terhubung bisa melakukan *backup* atau *failover* seperti pada gambar 6.



GAMBAR 6
(SKENARIO KEDUA PENGUJIAN *FAILOVER* JARINGAN)

2. Pengujian *Quality of Service (QoS)*

Pada pengujian ini, sistem diuji berdasarkan performa jaringan sesuai dengan standar *Telecommunications and Internet Protocol Harmonization Over Networks* (TIPHON) yang dikembangkan oleh *European Telecommunications Standards Institute* (ETSI), dengan parameter *throughput, delay, jitter dan packet loss,* guna memastikan efisiensi dan keandalan dalam pengelolaan trafik data [11], [12], [13].

a) Throughput

Throughput merupakan ukuran jumlah data yang berhasil ditransfer dalam suatu waktu tertentu, atau bisa diartikan sebagai bandwidth yang sebenarnya, yang diukur selama periode waktu tertentu saat file ditransfer. Untuk perhitungan dari *throughput* dapat dilihat pada persamaan dibawah ini:

$$\text{Throughput} = \frac{\text{Paket yang diterima}}{\text{Waktu}} \quad (1)$$

Hasil *throughput* yang telah didapatkan akan dibandingkan dengan ketentuan seperti pada tabel 1.

TABEL 1
(THROUGHPUT STANDAR TIPHON)

Kategori Throughput	Throughput	Indeks
Sangat Bagus	> 50 Mbps	4
Bagus	25-50 Mbps	3
Sedang	10-25 Mbps	2
Buruk	< 10	1

b) Delay

Delay adalah durasi yang diperlukan bagi data untuk melakukan perjalanan dari sumber ke tujuan. Faktor-faktor yang dapat memengaruhi delay meliputi jarak, jenis media fisik, kemacetan jaringan, serta waktu pemrosesan yang lama. Untuk perhitungan dari *delay* dapat dilihat pada persamaan dibawah ini:

$$\text{Delay} = \frac{\text{Total Delay}}{\text{Paket data yang diterima}} \quad (2)$$

Total Variasi Delay didapatkan dari rumus sebagai berikut :

$$(Delay\ 2 - Delay\ 1) + \cdots + (Delay\ n - Delay\ (n-1))$$

Hasil *delay* yang telah didapatkan akan dibandingkan dengan ketentuan seperti pada tabel 2.

TABEL 2
(DELAY STANDAR TIPHON)

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

c) Jitter

Jitter atau variasi delay, adalah selisih antara waktu *delay* paket pertama dan paket berikutnya. Variasi yang besar dalam *delay* dapat mempengaruhi kualitas data yang ditransmisikan. Toleransi *jitter* dalam jaringan dipengaruhi oleh kedalaman *buffer jitter* yang tersedia. semakin besar *buffer*, semakin baik jaringan dapat mengurangi efek *jitter*. Untuk perhitungan dari *jitter* dapat dilihat pada persamaan dibawah ini:

$$\text{Jitter} = \frac{\text{Total Variasi Delay}}{\text{Paket data yang diterima} - 1} \quad (3)$$

Hasil *jitter* yang telah didapatkan akan dibandingkan dengan ketentuan seperti pada tabel 3

TABEL 3
(JITTER STANDAR TIPHON)

Kategori Jitter	Jitter (ms)	Indeks
Sangat Bagus	0 ms	4
Bagus	0 ms s/d 75 ms	3
Sedang	75 ms s/d 125 ms	2
Buruk	125 s/d 225 ms	1

d) Packet Loss

Packet Loss adalah parameter yang menunjukkan jumlah total paket yang hilang selama transmisi data. Salah satu penyebab terjadinya *packet loss* adalah ketika antrian melebihi kapasitas *buffer* di setiap *node* jaringan. Untuk perhitungan dari *Packet Loss* terdapat pada persamaan berikut:

$$\text{Packet loss} = \left(\frac{\text{Paket yang dikirim} - \text{Paket yang diterima}}{\text{Paket yang dikirim}}\right) X\ 100 \quad (4)$$

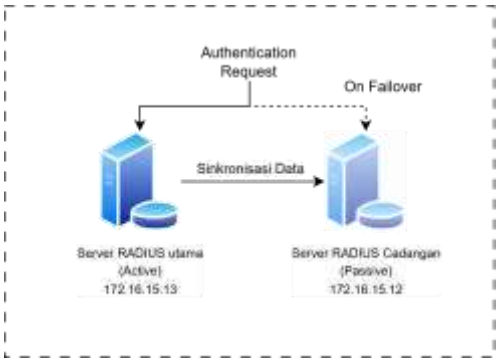
Hasil *packet loss* yang telah didapatkan akan dibandingkan dengan ketentuan seperti pada tabel 4.

TABEL 4
(PACKET LOSS STANDAR TIPHON)

Kategori Packet Loss	Packet Loss (%)	Indeks
Sangat Bagus	0	4
Bagus	≥ 3 %	3
Sedang	≥ 15%	2
Buruk	≥ 25%	1

3. Pengujian Failover Server RADIUS

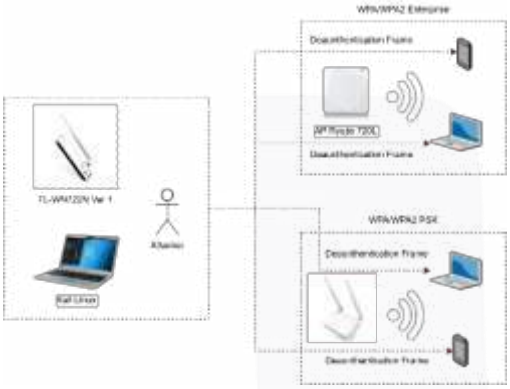
Pengujian ini bertujuan untuk memvalidasi sistem *failover* pada *Server* RADIUS dengan mensimulasikan kegagalan pada *server* utama untuk memastikan sistem dapat beralih secara otomatis ke *server* cadangan tanpa mengganggu layanan autentikasi. Untuk menjaga konsistensi data, dilakukan sinkronisasi *database user* setiap satu menit menggunakan fitur Rsync yang dijadwalkan secara otomatis oleh crontab. Mekanisme ini memastikan data pengguna, termasuk akun baru, selalu diperbarui pada *server* cadangan, sehingga meminimalkan risiko kehilangan data saat terjadi kegagalan permanen dan memvalidasi kelancaran proses *failover* secara keseluruhan. Untuk ilustrasinya seperti pada gambar 7



GAMBAR 7
(SKENARIO *FAILOVER SERVER* RADIUS)

4. Pengujian Keamanan Jaringan

Pengujian keamanan jaringan ini dilakukan dalam dua kondisi. Pertama, dilakukan simulasi serangan aktif menggunakan sistem operasi Kali Linux secara virtual dengan *network adapter* tambahan dalam *mode monitor* untuk menguji apakah sistem keamanan WPA/WPA2 Enterprise dapat mengatasi serangan nirkabel. Serangan yang disimulasikan adalah deautentikasi dan *evil twin attack*, dengan hasil analisis berupa penentuan keberhasilan serangan tersebut. Untuk simulasi penyerangan seperti pada gambar 8 dan 9.



GAMBAR 8
(SIMULASI SERANGAN DENGAN *DEAUTHENTICATION)*

GAMBAR 9
(SIMULASI SERANGAN DENGAN *EVIL TWIN ATTACK)*

Kondisi kedua adalah analisis kerentanan pada perangkat lunak FreeRADIUS berdasarkan daftar *Common Vulnerabilities and Exposures* (CVE) yang sesuai dengan versi yang digunakan, di mana skor risiko akan dihitung menggunakan rumus dari *National Institute of Standards and Technology* (NIST). Untuk perhitungan hasilnya dapat dilihat pada persamaan berikut:

$$\text{Risk} = \text{Threat Likelihood} \times \text{Vulnerability Severity} \quad (5)$$

5. Pengujian Kecepatan Autentikasi

Pada pengujian ini akan dilakukan pengujian kecepatan autentikasi di sisi klien. Pada pengujian ini 1 profile RADIUS akan dilakukan pengujian autentikasi sebanyak 30 kali dan hasil dari penelitian ini berupa rata-rata dari proses 30 kali pengujian autentikasi tersebut seperti persamaan dibawah ini

$$\text{Rata} - \text{Rata Autentikasi} = \frac{\text{T1} + \text{T2} + \cdots + \text{T30}}{30} \quad (3)$$

VII. HASIL DAN PEMBAHASAN

A. Load Balancing dan Failover Jaringan

Kinerja *load balancing* dinilai dengan membandingkan *throughput* dari setiap koneksi ISP secara individual terhadap *throughput* gabungan ketika PCC aktif. Hasilnya dirangkum pada tabel 5.

TABEL 5
(HASIL PENGUJIAN THROUGHPUT LOAD BALANCING JARINGAN)

Skenario Koneksi	Download Throughput (Mbps)	Upload Throughput (Mbps)
ISP 1 Only	88.32 Mbps	32.54 Mbps
ISP 2 Only	30.39 Mbps	14.66 Mbps
ISP 1 + ISP 2 (PCC)	66.29 Mbps	49.01 Mbps

Data dengan jelas menunjukkan bahwa penerapan PCC berhasil menyeimbangkan beban koneksi. Sistem *load balancing* PCC berhasil mengoptimalkan *bandwidth*, terutama meningkatkan *throughput upload* hingga 49,01 Mbps. Selain itu, mekanisme *failover* jaringan terbukti sangat andal, karena berhasil menangani keempat skenario simulasi gangguan. Kemudian pada pengujian mekanisme *failover* ini dilakukan dalam empat skenario untuk mengukur waktu respons dan keandalannya. Hasilnya, seperti yang dirinci pada tabel 6, menunjukkan bahwa sistem berhasil menangani setiap skenario kegagalan.

TABEL 6
(HASIL PENGUJIAN FAILOVER JARINGAN)

No.	Skenario Percobaan	Downtime (s)	Status
1	WAN 1 Diputus Kabel	3s	Berhasil
2	WAN 2 Diputus Kabel	2s	Berhasil
3	Interface ether1 Diputus	5s	Berhasil
4	Interface ether2 Diputus	2s	Berhasil

Sebagaimana ditunjukkan pada Tabel 6, reaksi sistem terhadap kegagalan terjadi secara langsung, dengan *downtime* yang sangat singkat, hanya berkisar antara 2 hingga 5 detik. Penundaan terlama selama 5 detik teramati ketika antarmuka utama (ether1) dinonaktifkan secara manual, yang merupakan perilaku yang wajar karena sistem perlu melakukan penyesuaian rute. Transisi yang cepat dalam semua skenario mengonfirmasi bahwa konfigurasi *failover* ini sangat efektif dan andal, memastikan kelangsungan layanan bagi pengguna hampir tanpa gangguan.

B. *Quality of Services* (QoS)

Untuk mengevaluasi performa jaringan saat beban operasional, dilakukan analisis Kualitas Layanan. Penilaian ini berfokus pada empat parameter utama yaitu *throughput*, *delay*, *jitter*, dan *packet loss*.

1. Throughput

Pada pengujian ini dapatkan dari percobaan selama satu minggu pada saat *traffic* sedang kondisi yang padat. Untuk hasil dari *throughput* yang didapatkan berdasarkan standar TIPHON seperti pada Tabel 7.

TABEL 7
(HASIL THROUGHPUT)

Hari	Throughput	Indeks	Kategori
1	54.42 Mbps	4	Sangat Bagus
2	66.4 Mbps	4	Sangat Bagus
3	21.8 Mbps	2	Sedang
4	10.71 Mbps	2	Sedang
5	55.41 Mbps	4	Sangat Bagus
6	32.2 Mbps	3	Bagus
7	56.6 Mbps	4	Sangat Bagus

Analisis *throughput* menunjukkan bahwa kinerja jaringan bersifat fluktuatif. Meskipun sistem berhasil mencapai kecepatan tertinggi hingga 66,4 Mbps, ketidakstabilan performa ini disebabkan oleh penggunaan koneksi *broadband* kelas rumahan yang bersifat *shared* (jalur dipakai bersama pengguna lain). Oleh karena itu, untuk menjamin performa yang stabil dan sesuai untuk lingkungan perusahaan, diperlukan peningkatan (*upgrade*) ke koneksi internet *dedicated* yang jalurnya khusus.

2. Delay

Pengujian dilakukan pada kondisi trafik tinggi maupun rendah dengan mengirimkan paket *Internet Control Message Protocol* (ICMP) ke berbagai domain. Hasilnya, yang dikategorikan menggunakan standar TIPHON, disajikan pada tabel 8.

TABEL 8
(HASIL DELAY)

No	Domain Target	Delay (Low Traffic)	Delay (High Traffic)	Indeks
1	google.com	25ms	34ms	4
2	detik.com	24ms	71ms	4
3	facebook.com	29ms	43ms	4
4	cloudflare.com	26ms	28ms	4
5	akamai.com	107ms	76ms	4

Nilai *delay* jaringan secara umum lebih rendah saat trafik sepi dibandingkan saat trafik padat. Namun, domain Akamai.com menunjukkan *delay* yang tinggi secara konsisten sekitar 107-109ms di kedua kondisi tersebut. Hal ini kemungkinan besar disebabkan oleh faktor eksternal seperti lokasi server yang jauh secara geografis dan rute jaringan *Content Delivery Network* (CDN) yang kompleks, bukan karena beban trafik.

3. Jitter

Pada hasil percobaan jitter akan dibandingkan berdasarkan standar TIPHON menggunakan dua skenario yang berbeda yaitu *high traffic* dan *low traffic* seperti pada tabel 9.

TABEL 9
(HASIL JITTER)

No	Domain Target	Jitter (Low Traffic)	Jitter (High Traffic)	Indeks
1	google.com	14ms	7ms	3
2	detik.com	5.9ms	43ms	3
3	facebook.com	24ms	8ms	3
4	cloudflare.com	1.56ms	20ms	3
5	akamai.com	5.82ms	12ms	3

Pengujian menunjukkan bahwa nilai *jitter* jauh lebih baik dan stabil saat trafik rendah dengan rata-rata 10.26ms dibandingkan saat trafik padat dengan rata-rata 30.2ms. Peningkatan *jitter* yang signifikan saat trafik padat disebabkan oleh antrian paket yang lebih lama, pembagian *bandwidth* untuk banyak koneksi, serta beban proses pada perangkat jaringan yang lebih tinggi. Kinerja CPU router yang kurang mumpuni juga bisa menjadi faktor penyebab.

4. Packet Loss

Pada hasil penelitian packet loss akan dibandingkan dengan standar TIPHON menggunakan dua skenario yang berbeda seperti pengujian throughput, delay dan jitter sebelumnya yaitu *high traffic* dan *low traffic*. Seperti pada tabel 10.

TABEL 10
(HASIL PACKET LOSS)

No	Domain Target	Packet Loss (Low Traffic)	Packet Loss (High Traffic)	Index
1	google.com	0%	2%	3
2	detik.com	0%	2%	3
3	facebook.com	0%	5%	3
4	cloudflare.com	0%	0%	4
5	akamai.com	8%	22%	2

Hasil penelitian menunjukkan bahwa trafik yang padat secara langsung menyebabkan hilangnya paket data (*packet loss*), sementara saat trafik sepi performa jaringan jauh lebih optimal. Pada kondisi trafik padat, hampir semua domain mengalami *packet loss* antara 2% hingga 11%, dengan Cloudflare.com menjadi yang terbaik karena berhasil mempertahankan 0% *packet loss*. Sebaliknya, saat trafik sepi, hampir semua domain juga mencapai 0% *packet loss*. Domain Akamai.com menjadi pengecualian utama karena secara konsisten mengalami *packet loss* tertinggi di kedua kondisi (11% saat padat dan 8% saat sepi), yang kemungkinan besar disebabkan oleh infrastruktur CDN dan arsitektur jaringannya yang unik sehingga menimbulkan masalah *routing* pada ISP tertentu.

C. Failover Server RADIUS

Pengujian Failover RADIUS berhasil memvalidasi proses sinkronisasi data dan *failover* otomatis. Otomatisasi rsync dengan cron secara efektif mereplikasi data pengguna baru ke server cadangan dalam waktu sekitar satu detik. Sementara itu, analisis Wireshark mengonfirmasi bahwa permintaan autentikasi dialihkan dengan benar ke server cadangan saat server utama mengalami gangguan, sehingga menjamin kelangsungan layanan. Waktu henti (*downtime*) yang terkait

dengan proses *failover* ini telah diukur dan dirinci pada tabel 11.

TABEL 11
(HASIL DARI FAILOVER SERVER RADIUS)

No	Failover Scenario	Downtime (s)	Status
1	Server Utama ke Server Cadangan	5s	Berhasil
2	Server Cadangan ke Server Utama	66s	Berhasil

D. Keamanan Jaringan

Keamanan jaringan dievaluasi melalui simulasi serangan, penilaian kerentanan. Konfigurasi WPA2-Enterprise menunjukkan ketahanan yang beragam; sistem ini rentan terhadap serangan deautentikasi, mirip dengan WPA2-PSK standar. Namun, selama serangan evil twin, sistem menunjukkan peningkatan keamanan kunci dibandingkan PSK dengan melindungi kata sandi pengguna, hanya mengekspos hash terenkripsi sementara nama pengguna tertangkap dalam teks jelas. Hal ini menegaskan pentingnya kata sandi pengguna yang kuat untuk mencegah hash tersebut dipecahkan dalam serangan *dictionary offline*.

Penilaian kerentanan juga dilakukan pada versi FreeRADIUS 3.0.21 yang digunakan dengan cara mencocokkannya ke basis data *Common Vulnerabilities and Exposures* (CVE). Analisis mengidentifikasi tiga kerentanan yang diketahui, dengan satu dinilai memiliki tingkat risiko 'Kritis', seperti dirinci dalam Tabel 12. Temuan ini menegaskan pentingnya kritis dari pembaruan dan pemeliharaan perangkat lunak secara berkelanjutan untuk mengurangi risiko keamanan yang melekat.

TABEL 12
(HASIL KERENTANAN BERDASARKAN CVE)

Kerentanan	CVE ID	Threat Likelihood	Vulnerability Severity	Risk Score	Risk Level
BlastRADIUS	CVE-2024-3596	4	5	20	Kritis
Malformed Binary Attribute	CVE-2022-41861	2	3	6	Rendah
EAP-SIM NULL Pointer	CVE-2022-41860	2	4	8	Sedang

E. Kecepatan Autentikasi

Pada pengujian ini mengukur berapakah kecepatan yang didapatkan untuk 1 user pada proses autentikasi. Hasil dari pengukuran tersebut terdapat pada tabel 13.

TABEL 13
(HASIL DARI KECEPATAN AUTENTIKASI SERVER RADIUS)

No	User	Rata-Rata (ms)
1	User 1	0.62 ms
2	User 2	0.68 ms
3	User 3	1.25 ms
4	User 4	0.78 ms
5	User 5	0.56 ms
6	User 6	0.83 ms
7	User 7	0.90 ms
8	User 8	0.80 ms

Berdasarkan pengujian kecepatan autentikasi yang hasilnya diukur dalam milidetik, dapat dilihat bahwa proses masuk pengguna secara keseluruhan sangatlah cepat dan efisien, di mana waktu tercepat yang tercatat adalah 0,56 ms oleh User Ke-5 dan yang terlambat adalah 1,25 ms oleh User Ke-3, yang artinya bahkan proses paling lambat sekalipun tetap selesai dalam waktu yang jauh lebih singkat.

VIII. KESIMPULAN

Berdasarkan pembahasan hasil penelitian ini didapatkan bahwa sistem ini telah berhasil menjawab rumusan masalah dengan mengimplementasikan sebuah sistem jaringan enterprise yang terpusat. Implementasi Server RADIUS dengan keamanan WPA2 Enterprise terbukti mampu menyediakan sistem autentikasi yang kuat dan efisien. Hal ini ditunjukkan dengan kecepatan autentikasi rata-rata 0,8 ms per pengguna serta kemampuannya melindungi kredensial dari upaya pencurian kata sandi. Dari sisi performa dan keandalan, metode Load Balancing PCC berhasil meningkatkan throughput unduh gabungan menjadi 66,29 Mbps dan unggah gabungan menjadi 49,01 Mbps. Keandalan sistem juga didukung oleh hasil pengujian *Quality of Service* (QoS) yang menunjukkan parameter delay, jitter, dan packet loss berada dalam kategori "Sangat Bagus" atau "Bagus" pada kondisi trafik normal. Mekanisme failover ganda melengkapi keandalan ini, dengan waktu peralihan koneksi jaringan rata-rata hanya 3 detik dan layanan autentikasi yang tetap terjaga saat server utama gagal. Meskipun ditemukan penurunan kinerja QoS pada trafik padat, secara keseluruhan sistem ini terbukti menjadi solusi yang efektif dan stabil.

REFERENSI

[1] C. A. Ochoa Villanueva and A. Roman-Gonzalez, “Implementation of a RADIUS Server for Access Control through Authentication in Wireless Networks,” *International Journal of Advanced and Applied Sciences*, vol. 10, no. 3, pp. 183–188, Mar. 2023, doi: 10.21833/ijaas.2023.03.022.

[2] E. R. Amalia, Nurheki, R. Saputra, C. Ramadhana, and E. H. Yossy, “Computer Network Design and Implementation Using Load Balancing Technique with Per Connection Classifier (PCC) Method Based on MikroTik Router,” *Procedia Comput Sci*, vol. 216, pp. 103–111, Jan. 2023, doi: 10.1016/J.PROCS.2022.12.116.

[3] T. Rahman, E. Sulistianto, A. Sudibyo, S. Sumarna, and B. Wijonarko, “Per Connection Classifier Load Balancing dan Failover MikroTik pada Dua Line Internet,” *JIKA (Jurnal Informatika)*, vol. 5, no. 2, pp. 195–209, Jun. 2021, doi: 10.31000/JIKA.V5I2.4517.

[4] H. A. Hadi, Iin, G. Dwilestari, A. Faqih, and N. D. Nuris, “Manajemen Autentifikasi User Menggunakan Metode Radius Server pada RS Jantung Hasna Medika,” *KOPERTIP : Scientific Journal of Informatics Management and Computer*, vol. 6, no. 2, pp. 34–41, Jun. 2022, doi: 10.32485/KOPERTIP.V6I2.133.

[5] R. Fahrizal, M. I. Santoso, and M. Z. Arifin, “Implementation Multipath Routing with Equal Cost Multipath (ECMP) and Per Connection Classifier (PCC),” *Proceeding - 2020 2nd International Conference on Industrial Electrical and Electronics,*

ICIEE 2020, pp. 169–173, Oct. 2020, doi: 10.1109/ICIEE49813.2020.9277496.

[6] K. A. T. Indah and I. N. K. Wardana, "The Implementation of Radius Server for WiFi Pass Using the Mechanism of Access Point Controller in Department of Electrical Engineering Building, Bali State Polytechnic," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Mar. 2020. doi: 10.1088/1742-6596/1450/1/012073.

[7] M. M. Kovtsur, A. Muthanna, H. M. R. Al-Khafaji, P. Karelsky, A. Kozmyan, and G. Voroshnin, "IPTV Access Methods with RADIUS-Server Authorization," *Journal of Information Technology Management*, vol. 14, no. 2, pp. 80–89, 2022, doi: 10.22059/jitm.2022.86929.

[8] A. D. Yudhistira and R. Harwahyu, "Implementation Strategy Analysis of Network Security using dalo RADIUS and Pi-hole DNS Server to enhance Computer Network Security, Case Study: XYZ as a Fintech Company," *Jurnal Indonesia Sosial Teknologi*, vol. 5, no. 10, pp. 4364–4379, Oct. 2024, doi: 10.59141/JIST.V5I10.5321.

[9] I. P. Saputra, R. Yusuf, and U. Saprudin, "Implementasi Cloud Computing sebagai Radius Server pada Jaringan Internet Router MikroTik," *Journal Computer Science and Information Systems : J-Cosys*, vol. 1, no. 2, pp. 81–86, Jul. 2021, doi: 10.53514/JC.V1I2.67.

[10] A. Sumiah, F. Nugraha, and A. Permana, "Penerapan Teknologi VPN L2TP dan Protokol AAA RADIUS pada Jaringan Hotspot," *NUANSA INFORMATIKA*, vol. 17, no. 2, pp. 154–160, 2023, doi: 10.25134/NUANSA.

[11] Idham, Rodianto, and H. Wahyudi, "Implementasi Load Balancing dan Failover pada Jaringan Internet Menggunakan Metode NTH," *Jurnal Informatika, Teknologi dan Sains*, vol. 4, no. 3, pp. 131–136, Aug. 2022, doi: 10.51401/JINTEKS.V4I3.1904.

[12] I. S. N. Nisa, Rahmat Miyarno Saputro, Tegar Fatwa Nugroho, and Alfirna Rizqi Lahitani, "Analisis Quality of Service (QoS) Menggunakan Standar Parameter Tiphon pada Jaringan Internet Berbasis Wi-Fi Kampus 1 Unjaya," *Teknomatika: Jurnal Informatika dan Komputer*, vol. 17, no. 1, pp. 1–9, Apr. 2024, doi: 10.30989/teknomatika.v17i1.1307.

[13] S. Turangga, Martanto, and Y. A. Wijaya, "Analisis Internet Menggunakan Parameter Quality of Service pada Alfamart Tuparev 70," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 6, no. 1, pp. 392–398, Apr. 2022, doi: 10.36040/JATI.V6I1.4693.