JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

ANALISIS DAN IMPLEMENTASI MULTI-FACTOR AUTHENTICATION (MFA) UNTUK MENCEGAH SERANGAN BRUTE FORCE PADA SSH MIKROTIK

Eureka Diaandisy*[1)], Diah Risqiwati[2)]
1. Informatika, Teknik, Universitas Muhammadiyah Malang, Indonesia
2. Informatika, Teknik, Universitas Muhammadiyah Malang, Indonesia

Article Info
Kata Kunci: Brute force; Keamanan jaringan; MikroTik; Multi-Factor Authentication; SSH

Keywords: Brute force; MikroTik; Multi-Factor Authentication; Network security; SSH

Article history:
Received 1 October 2025
Revised 11 November 2025
Accepted 17 November 2025
Available online 1 December 2025

DOI :
https://doi.org/10.29100/jipi.v10i4.9234

* Corresponding author.
Corresponding Author
E-mail address:
eurekadiaandisy8@webmail.umm.ac.id

ABSTRAK
Keamanan SSH pada perangkat Mikrotik sangat penting mengingat risiko serangan brute force yang dapat merusak sistem jaringan. Penelitian ini bertujuan menguji efektivitas penerapan Multi-Factor Authentication (MFA) dalam mencegah serangan brute force pada SSH MikroTik RouterOS. MFA diimplementasikan menggunakan One-Time Password (OTP) melalui RADIUS dan User Manager. Metode pengujian melibatkan simulasi serangan brute force menggunakan THC-Hydra pada dua skenario: sistem tanpa MFA dan sistem dengan MFA. Hasil pengujian menunjukkan bahwa pada sistem tanpa MFA serangan berhasil mendapatkan kredensial user (username dan password) dalam waktu sekitar 335 detik, sedangkan pada sistem dengan menggunakan MFA serangan gagal mendapatkan kredensial sehingga terjadi kegagalan login berulang dalam durasi 332 detik. Selain itu, penggunaan CPU dan bandwidth relatif stabil pada kedua skenario, dengan pemakaian dan rata-rata CPU yaitu maksimal 14% dan 2%, sedangkan pada bandwidth sekitar 9,41–9,59 Kb. Temuan ini menegaskan bahwa penerapan MFA dapat meningkatkan keamanan SSH MikroTik secara signifikan tanpa menimbulkan beban sistem yang berarti. Penelitian ini merekomendasikan penerapan MFA sebagai lapisan keamanan tambahan pada lingkungan jaringan produksi untuk mencegah serangan brute force. Hasil penelitian diharapkan menjadi dasar pengembangan metode autentikasi canggih seperti berbasis biometrik atau FIDO di masa mendatang.

ABSTRACT
SSH security on MikroTik devices is crucial due to the risk of brute force attacks that can compromise network systems. This study aims to evaluate the effectiveness of implementing Multi-Factor Authentication (MFA) in preventing brute force attacks on SSH MikroTik RouterOS. MFA is implemented using a One-Time Password (OTP) through RADIUS and User Manager. The testing method involves simulating brute force attacks using THC-Hydra under two scenarios: a system without MFA and a system with MFA. The results show that in the system without MFA, the attack successfully obtained user credentials (username and password) in approximately 335 seconds. In contrast, in the system with MFA, the attack failed to retrieve credentials, resulting in repeated login failures over a period of 332 seconds. Additionally, CPU and bandwidth usage remained relatively stable in both scenarios, with CPU usage peaking at 14% and averaging 2%, and bandwidth usage ranging from approximately 9.41 to 9.59 Kb. These findings confirm that MFA implementation significantly enhances SSH security on MikroTik without imposing a significant system burden. This study recommends the adoption of MFA as an additional security layer in production network environments to prevent brute force attacks. The results are expected to serve as a foundation for developing advanced authentication methods such as biometric or FIDO-based authentication in the future.

.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

I. PENDAHULUAN

DALAM konteks keamanan jaringan, transformasi digital tidak hanya terbatas pada adopsi perangkat lunak dan perangkat keras, tetapi juga mencakup pengelolaan serta perlindungan data dari ancaman siber yang semakin kompleks [1]. Serangan *brute force* merupakan metode serangan yang mencoba kombinasi login secara berulang-ulang hingga berhasil menembus sistem [2]. Serangan ini krusial karena dapat menyebabkan kebocoran data, pengambil alihan kontrol perangkat, dan berbagai kerugian operasional pada organisasi yang tidak memiliki perlindungan autentikasi yang memadai [3]. Dalam penelitian Aji dkk. melaporkan tercatat 259.646 percobaan brute force pada server web universitas dalam suatu periode pengamatan [4]. Pada tingkat global, *Data Breach Investigations Report* (DBIR) 2025 menunjukkan *credential abuse* (penyalahgunaan kredensial) meningkat 34% dibanding tahun sebelumnya [5], menegaskan tren serangan berbasis kata sandi yang semakin marak. Kondisi ini memperkuat urgensi perlindungan autentikasi SSH di organisasi pendidikan dan organisasi.

MikroTik RouterOS merupakan sistem operasi berbasis Linux yang dirancang khusus untuk menjadikan komputer sebagai router jaringan yang handal. Sistem ini dilengkapi berbagai fitur seperti manajemen bandwidth, firewall, hotspot, serta routing, dan telah banyak digunakan oleh ISP, penyedia hotspot, serta pemilik warnet karena kemampuannya dalam menangani jaringan kabel maupun nirkabel secara efisien [6]. Sebagai perangkat jaringan yang populer, terutama karena fleksibilitasnya dalam konfigurasi dan harga yang terjangkau, menjadi salah satu sasaran utama serangan brute force melalui protokol SSH. Serangan brute force terhadap SSH dapat membuka akses tidak sah, mengakibatkan perubahan konfigurasi, pencurian data, bahkan pengendalian penuh atas jaringan [7].

Secure Shell (SSH) adalah protokol berbasis TCP/IP yang menyediakan komunikasi jaringan yang aman melalui enkripsi dan autentikasi yang kuat. SSH menjamin bahwa identitas pengguna dapat diverifikasi secara andal, data yang dikirim tetap rahasia, dan integritas pesan terjaga, sehingga SSH menjadi solusi penting dalam menjaga privasi dan keamanan koneksi jarak jauh antar perangkat di jaringan [8].

Pemilihan SSH sebagai fokus penelitian didasarkan pada tiga faktor utama. Pertama, SSH merupakan protokol standar untuk administrasi jarak jauh router MikroTik secara aman, sehingga menjadi target potensial bagi penyerang. Fauzi dkk. menegaskan bahwa protokol FTP dan SSH pada router MikroTik rentan terhadap serangan brute force yang dapat menyebabkan akses tidak sah dan pencurian data [9]. Kedua, dalam mode interaktif SSH setiap ketukan tombol pengguna dikirim dalam paket IP terpisah segera setelah ditekan, sehingga penyerang dapat mengamati waktu antar ketukan (*inter-keystroke timing*) dari pola kedatangan paket. Dengan memanfaatkan informasi *timing* tersebut dan teknik statistik sederhana, penyerang bahkan dapat mengungkap panjang *password* dan pola pengetikan pengguna dari sesi SSH [10]. Selain itu, jika *host key* SSH tidak tervalidasi dengan benar, koneksi rentan disusupi *man-in-the-middle* (MITM) yang dapat mencuri kombinasi *username*/*password* pengguna [11]. Ketiga, Bäumer dkk. membahas *Terrapin Attack* yang memanfaatkan manipulasi nomor urut paket pada protokol SSH. Mereka menemukan bahwa SSH tidak mereset *counter* urutan pesan ketika kunci enkripsi diaktifkan, sehingga penyerang dapat menambahkan atau menghapus paket di awal saluran SSH tanpa terdeteksi. Dengan cara ini, integritas saluran SSH dapat dilanggar (*prefix truncation*), misalnya untuk menonaktifkan mekanisme keamanan baru atau menurunkan algoritma kriptografi tanpa diketahui [12].

Untuk itu, penelitian ini mengusulkan penerapan sistem Multi-Factor Authentication (MFA) sebagai solusi preventif yang lebih efektif. MFA adalah metode autentikasi yang mengharuskan pengguna untuk melewati lebih dari satu tahapan verifikasi, seperti memasukkan password dan kode verifikasi dari aplikasi atau perangkat tertentu. Kajian internasional terkait MFA di perangkat jaringan semakin berkembang. Sebagai contoh, sistem operasi Cisco IOS XR kini mendukung integrasi token MFA pada proses *login* SSH, mengkombinasikan autentikasi *password* dengan faktor kriptografis tambahan [13]. Metode ini terbukti secara signifikan menurunkan tingkat keberhasilan serangan brute force karena menambahkan lapisan verifikasi yang tidak dapat ditembus hanya dengan kombinasi username dan password [14]. Oleh karena itu, penerapan MFA pada SSH menjadi penting untuk menutup celah autentikasi yang belum dapat teratasi dengan cara konvensional.

Upaya dalam mengatasi serangan brute force telah dilakukan dalam berbagai penelitian, seperti dengan menggunakan firewall filter, intrusion prevention system (IPS), dan port knocking. Namun solusi tersebut pada umumnya masih memiliki celah, baik karena pendekatannya yang reaktif atau tidak berfokus pada peningkatan lapisan autentikasi pengguna [15]. Dalam penelitian yang dilakukan oleh Ansharullah dkk., penggunaan IPS dapat membantu mendeteksi dan memblokir brute force terhadap SSH pada MikroTik, tetapi belum menyentuh aspek autentikasi ganda yang lebih aman [16]. Begitu pula dalam penelitian oleh Fauzi dkk., ditemukan bahwa penerapan CAPTCHA dan pembatasan login saja belum cukup untuk menghadang serangan yang terstruktur dan masif [9].

Dalam penelitian ini, sistem MFA akan diterapkan secara tidak langsung melalui RADIUS dan User Manager, sehingga akses ke SSH hanya dimungkinkan setelah pengguna berhasil melalui tahapan MFA. Penelitian ini dilakukan melalui simulasi pada PNETLab, sebuah platform virtualisasi jaringan yang memungkinkan pengujian

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

sistem keamanan tanpa harus menggunakan infrastruktur fisik secara langsung. Hal ini memungkinkan peneliti untuk mereplikasi kondisi nyata dan mengamati efektivitas sistem MFA yang diimplementasikan pada SSH MikroTik terhadap serangan brute force.

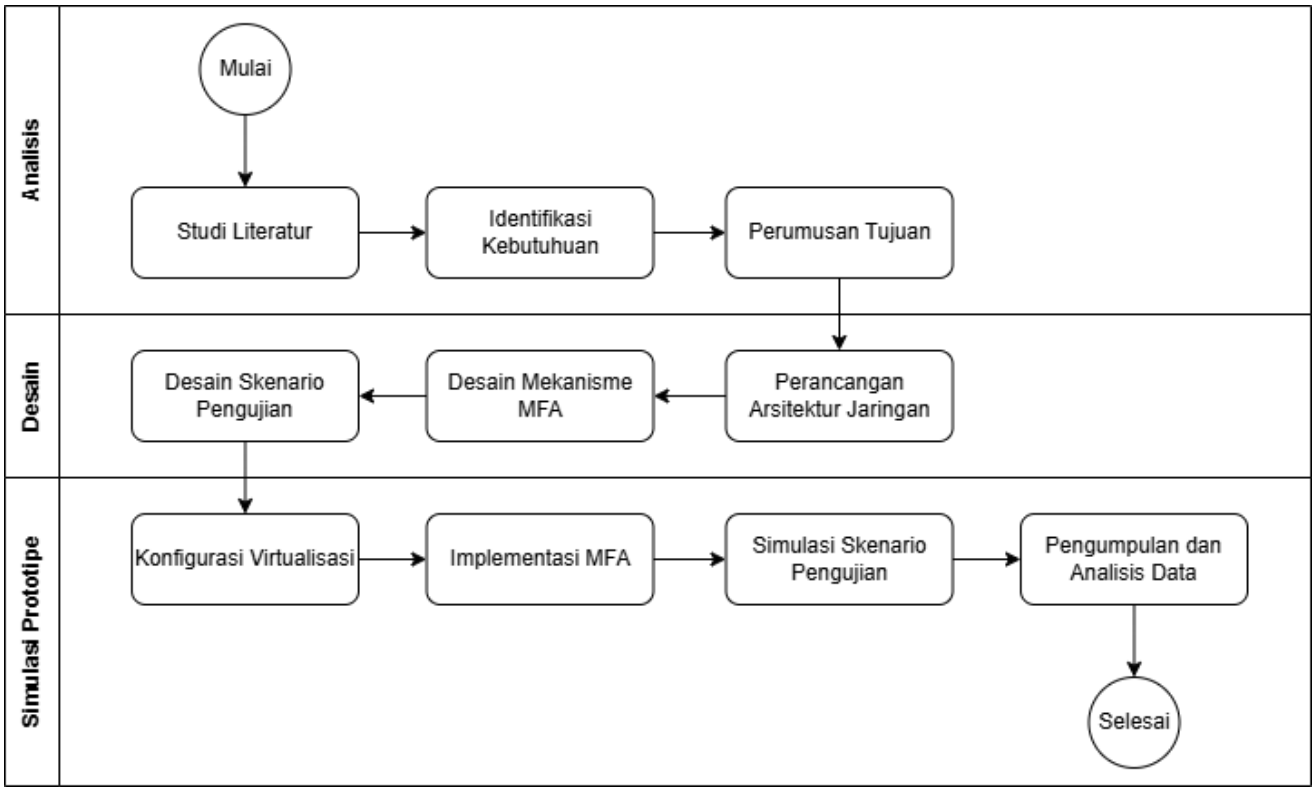
Penelitian ini menggunakan pendekatan NDLC (Network Development Life Cycle) yang terdiri dari tahapan-tahapan analisis, desain, implementasi, simulasi, monitoring, dan manajemen. Metode ini dipilih karena sangat relevan dalam pengembangan sistem jaringan dengan pendekatan yang terstruktur dan berorientasi pada solusi berkelanjutan [17].

Penelitian-penelitian sebelumnya telah memberikan kontribusi dalam penguatan sistem keamanan jaringan, namun umumnya tidak memberikan solusi pada level autentikasi pengguna. Sebagai contoh, penelitian oleh Yudi Mulyanto menunjukkan bahwa login MikroTik sangat rentan terhadap serangan brute force dan hanya menganalisis celah tanpa memberikan solusi autentikasi tambahan [7]. Sementara itu, studi oleh Setyowibowo dkk. mengusulkan metode simple port knocking, yang mampu menyembunyikan port akses, namun masih rentan terhadap serangan replay dan latency [15]. Dalam konteks ini, penelitian ini memiliki keunggulan berupa penerapan autentikasi ganda yang dapat melindungi sistem dari serangan brute force dengan pendekatan preventif dan user-centered.

Dengan demikian, kontribusi utama dari penelitian ini adalah penerapan solusi berbasis Multi-Factor Authentication pada SSH MikroTik, yang belum banyak dibahas dalam literatur sebelumnya. Fokus pada implementasi autentikasi berlapis ini diharapkan memberikan perlindungan lebih baik terhadap serangan brute force serta meningkatkan keandalan sistem jaringan di lingkungan organisasi. Penelitian ini juga memberikan kontribusi praktis berupa panduan teknis yang dapat diadaptasi oleh administrator jaringan dalam memperkuat sistem keamanannya.

Namun, penelitian ini memiliki keterbatasan, antara lain pada cakupan implementasi yang hanya dilakukan pada skenario berbasis simulasi, serta jenis autentikasi ganda yang masih bergantung pada perangkat pihak ketiga. Penelitian lanjutan dapat dikembangkan untuk memperluas jenis autentikasi dan integrasi langsung ke sistem SSH secara lebih native.

II. METODE PENELITIAN



Gambar 1. Alur Penelitian efektivitas MFA dengan NDLC

A. Metode NDLC dan Tahapan Penelitian

Penelitian ini menggunakan metode NDLC (*Network Development Life Cycle*) yang terdiri dari tiga tahap utama: analisis, desain, dan simulasi prototipe. Pemilihan metode ini didasarkan pada kesesuaiannya untuk pengembangan sistem keamanan jaringan berbasis virtualisasi, sebagaimana diterapkan dalam penelitian Naim dkk. untuk analisis QoS jaringan kabel dan nirkabel [18]. NDLC menekankan fase analisis kebutuhan, perancangan, prototyping,

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

implementasi, dan pemantauan jaringan secara berkelanjutan. Pendekatan ini memudahkan penelitian konfigurasi jaringan (termasuk pengaturan SSH dan RADIUS) serta evaluasi berulang hingga sistem memenuhi kriteria keamanan. Dibandingkan SDLC umum (fokus pada pengembangan perangkat lunak) atau PPDIOO Cisco (lebih linier untuk proyek infrastruktur besar), NDLC memberikan fleksibilitas prototyping jaringan yang lebih dinamis.

1) Analisis

Sesuai dengan *Gambar 1*, tahap pertama merupakan tahap Analisis. Pada tahap analisis, penelitian difokuskan pada identifikasi permasalahan utama yang menjadi dasar pengembangan sistem [19], yaitu kerentanan SSH pada perangkat MikroTik terhadap serangan brute force. Proses analisis dilakukan melalui studi literatur yang mencakup telaah terhadap serangan brute force pada SSH, kelemahan sistem autentikasi berbasis password tunggal, serta efektivitas metode Multi-Factor Authentication (MFA) dalam meningkatkan keamanan akses SSH. Berdasarkan studi literatur dan pengamatan awal, diidentifikasi bahwa sistem autentikasi pada MikroTik perlu ditingkatkan dengan menambahkan lapisan keamanan berupa MFA, khususnya melalui integrasi RADIUS dan OTP. Dari hasil identifikasi tersebut, dirumuskan tujuan penelitian, yaitu merancang dan mensimulasikan sistem MFA pada SSH MikroTik yang dapat meminimalisir risiko serangan brute force.

2) Desain

Tahap desain bertujuan menghasilkan rancangan solusi yang dapat menjawab permasalahan yang telah diidentifikasi pada tahap analisis. Desain dilakukan secara konseptual dan teknis, yang mencakup perancangan arsitektur jaringan dengan menggunakan PNETLab, di mana topologi virtual terdiri dari Router MikroTik sebagai target, server RADIUS untuk autentikasi, dan mesin Ubuntu sebagai simulasi penyerang, serta dirancang sedemikian rupa agar lingkungan uji tetap terisolasi dari jaringan nyata. Selain itu, pada tahap ini juga dilakukan perancangan mekanisme MFA, yaitu integrasi autentikasi dua faktor pada SSH MikroTik, dengan faktor pertama berupa username dan password, serta faktor kedua berupa OTP yang dihasilkan oleh aplikasi authenticator dan diverifikasi melalui RADIUS. Peneliti juga menyusun skenario pengujian yang meliputi akses SSH tanpa MFA sebagai baseline, akses SSH dengan MFA, serta simulasi serangan brute force, dengan parameter evaluasi tingkat keberhasilan serangan, waktu autentikasi, dan utilisasi sumber daya perangkat. Terakhir, peneliti menetapkan desain parameter evaluasi dengan menentukan metrik.

3) Simulasi Prototipe

Tahap simulasi prototipe merupakan implementasi dari desain yang telah dibuat dalam lingkungan virtual dengan tujuan utama untuk menguji efektivitas rancangan sistem sebelum diterapkan pada lingkungan nyata. Proses ini diawali dengan konfigurasi virtualisasi, di mana peneliti membangun simulasi jaringan di PNETLab dengan mendeploy node MikroTik, Ubuntu sebagai mesin penyerang, serta mengatur jaringan internal sesuai dengan desain yang telah dirancang. Selanjutnya, dilakukan implementasi MFA melalui konfigurasi dasar pada MikroTik. Setelah itu, peneliti menjalankan skenario pengujian tanpa MFA, dan dengan MFA. Data hasil simulasi dikumpulkan, kemudian dilakukan analisis untuk membandingkan efektivitas sistem sebelum dan sesudah penerapan MFA.

B. Arsitektur Jaringan

Gambar 2. Arsitektur Jaringan Virtualisasi menggunakan PNETLab

Penelitian ini mengimplementasikan arsitektur jaringan dalam lingkungan virtualisasi menggunakan PNETLab. Donaldson dkk. (2017) menyatakan bahwa virtualisasi melibatkan isolasi sistem operasi di dalam mesin virtual, sehingga memungkinkan pengujian keamanan pada lingkungan virtual tanpa mempengaruhi sistem fisik secara

langsung [20]. Seperti terlihat pada *Gambar 2*, sistem dirancang dengan model client-server tertutup dalam lingkungan virtual. Komponen utama arsitektur jaringan virtualisasi PNETLab pada *Gambar 2* meliputi:
1) MikroTik RouterOS, sebagai gateway dan RADIUS server.
2) Penyerang dengan menggunakan Ubuntu 22.04.
3) Jaringan Internal: Subnet 172.11.11.0/28 untuk isolasi traffic.

C. Tools Penelitian

1) PNETLab: Platform virtualisasi jaringan untuk simulasi.
2) THC-Hydra: THC-Hydra adalah sebuah perangkat lunak yang digunakan untuk melakukan penetrasi pada sistem *[21]*.

D. Parameter Evaluasi

Penelitian ini mengadopsi metrik evaluasi yang diusulkan oleh Yopi (2023), metrik tersebut mencakup aspek proteksi, usability, dan kinerja sistem [22]. Dengan modifikasi sesuai konteks keamanan:
1) Percobaan login.
2) Durasi Serangan.
3) Max CPU Usage
4) Average CPU Usage
5) Max Bandwidth (Kb)
6) Average Bandwidth (Kb)

E. Skenario Pengujian

Pada pengujian *brute force* digunakan tool THC-Hydra dalam mode *dictionary attack* dengan perintah *hydra -L userlist.txt -P passlist.txt ssh://172.11.11.1*, di mana opsi -L dan -P membaca daftar *username* dan *password* secara terpisah sehingga menghasilkan semua pasangan kombinasi. Pada penelitian ini, *userlist*.txt berisi 16 entri dan *passlist*.txt berisi 12. Semua pengaturan lain mengikuti nilai *default* pada versi Hydra yang digunakan, sehingga tidak ada opsi eksplisit seperti -t atau -w pada perintah penelitian, pendekatan ini dipilih agar fokus pengujian adalah efektivitas MFA, bukan optimasi serangan, sehingga semua kombinasi diuji hingga habis dan durasi serangan menjadi fungsi dari jumlah kombinasi dan kondisi jaringan pada saat pengujian. Skenario pengujian dirancang untuk mengukur efektivitas penerapan Multi-Factor Authentication (MFA) dalam mencegah serangan brute force pada layanan SSH MikroTik di lingkungan virtualisasi. Rincian skenario pengujian sebagai berikut:

1) Persiapan Lingkungan Uji

Persiapan lingkungan uji dilakukan dengan membangun topologi jaringan virtual menggunakan PNETLab, yang terdiri atas MikroTik RouterOS sebagai target utama, Ubuntu Desktop sebagai mesin penyerang, serta jaringan internal dengan rentang IP 172.11.11.0/28. Pada tahap ini, konfigurasi dasar pada MikroTik mencakup pengaktifan layanan SSH, pengaturan DHCP untuk manajemen alamat IP, serta integrasi RADIUS dan OTP guna mendukung penerapan MFA. Mekanisme OTP dalam penelitian ini diimplementasikan melalui server RADIUS User Manager Mikrotik. OTP dihasilkan berdasarkan standar OATH, menggunakan shared secret dan faktor waktu (TOTP) untuk menghasilkan kode numerik sekali-pakai. M'Raihi dkk. menjelaskan bahwa algoritma TOTP menggunakan input *shared secret* dan waktu sistem saat ini sehingga setiap kode berlaku hanya dalam periode singkat (secara *default* 30 detik) [23]. Server RADIUS melakukan verifikasi dengan menghitung ulang kode OTP menggunakan kunci dan waktu yang sama, serta membandingkannya dengan kode yang dimasukkan pengguna. Oleh karena itu, verifikasi OTP bergantung pada sinkronisasi waktu dan *shared secret*, menjamin bahwa hanya pemegang token perangkat (*authenticator*) yang dapat menghasilkan kode yang valid.

2) Skenario Pengujian Tanpa MFA

Skenario pengujian tanpa MFA dilakukan dengan mengaktifkan layanan SSH pada MikroTik tanpa penerapan mekanisme autentikasi ganda, sehingga hanya mengandalkan kombinasi username dan password. Selanjutnya, mesin penyerang berbasis Ubuntu Desktop digunakan untuk melakukan simulasi serangan brute force menggunakan tools THC-Hydra yang ditujukan ke layanan SSH MikroTik. Parameter yang diamati dalam skenario ini meliputi jumlah upaya login yang dilakukan, waktu yang dibutuhkan untuk mendapatkan akses, serta tingkat keberhasilan serangan.

3) Skenario Pengujian Dengan MFA

Skenario pengujian dengan MFA dilaksanakan dengan mengaktifkan layanan SSH pada MikroTik yang telah dikonfigurasi menggunakan autentikasi multi-faktor, yaitu kombinasi antara username, password, dan OTP yang dihasilkan oleh aplikasi authenticator. Mesin penyerang kembali menjalankan simulasi brute force menggunakan

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

tools THC-Hydra. Dalam pengujian ini, parameter yang diamati mencakup jumlah upaya login, waktu autentikasi, tingkat keberhasilan serangan, serta respon sistem terhadap percobaan login yang tidak valid.

4) Pengukuran dan Analisis

Pengukuran dan analisis dilakukan dengan mencatat seluruh hasil pengujian dari skenario tanpa MFA dan dengan MFA, kemudian dilakukan perbandingan untuk menilai efektivitas implementasi MFA. Analisis difokuskan pada sejauh mana penerapan MFA berdampak terhadap peningkatan keamanan dan perubahan kinerja sistem secara keseluruhan.

III. HASIL DAN PEMBAHASAN

Konfigurasi dilakukan dalam lingkungan simulasi menggunakan PNETLab, dengan fokus pada dua komponen utama: Router MikroTik sebagai target serangan dan Ubuntu Desktop sebagai mesin penyerang. Tujuan konfigurasi ini adalah untuk menyiapkan sistem secara realistis sesuai dengan skenario pengujian brute force sebelum dan sesudah implementasi Multi-Factor Authentication (MFA).

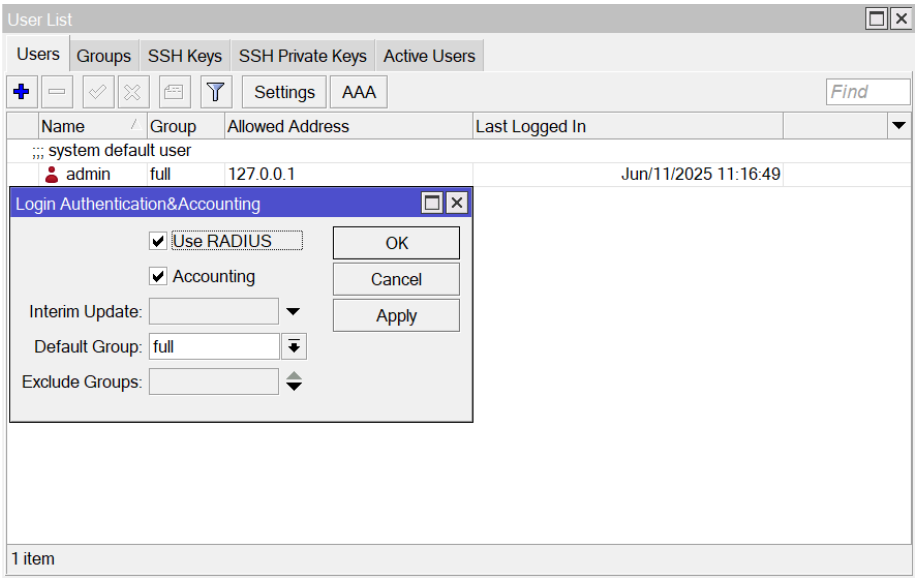
A. Konfigurasi

1) Konfigurasi MikroTik RouterOS (Versi 7.14) sebagai target utama.

Router MikroTik dikonfigurasi agar mendukung autentikasi berbasis RADIUS dan OTP. Pengaturan ini dilakukan dengan mengaktifkan layanan login berbasis RADIUS, menambahkan server lokal (127.0.0.1) sebagai RADIUS server, serta mengintegrasikan User Manager sebagai server autentikasi internal. Selain itu, akun pengguna mikrotik dibuat dengan atribut OTP (One-Time Password) agar mendukung metode autentikasi dua faktor.

Langkah-langkah konfigurasi meliputi:

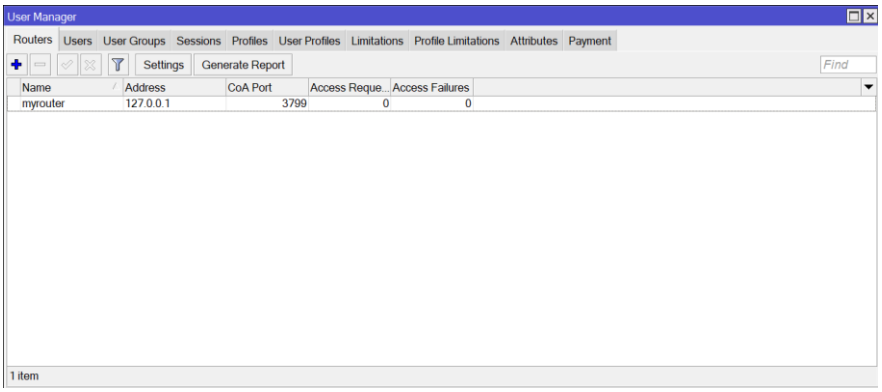
a. Mengaktifkan autentikasi RADIUS untuk login pengguna, dan menetapkan default group pengguna yang lolos autentikasi RADIUS sebagai full (akses penuh)



Gambar 3. Hasil konfigurasi *service User List Users*

b. Menambahkan server RADIUS lokal (127.0.0.1) sebagai tempat autentikasi untuk layanan login (SSH, Winbox, dll), dengan shared-secret.
c. Menambahkan MikroTik sebagai router klien ke dalam User Manager, agar User Manager bisa mengenali dan melayani permintaan autentikasi dari router tersebut menggunakan shared-secret yang sama.

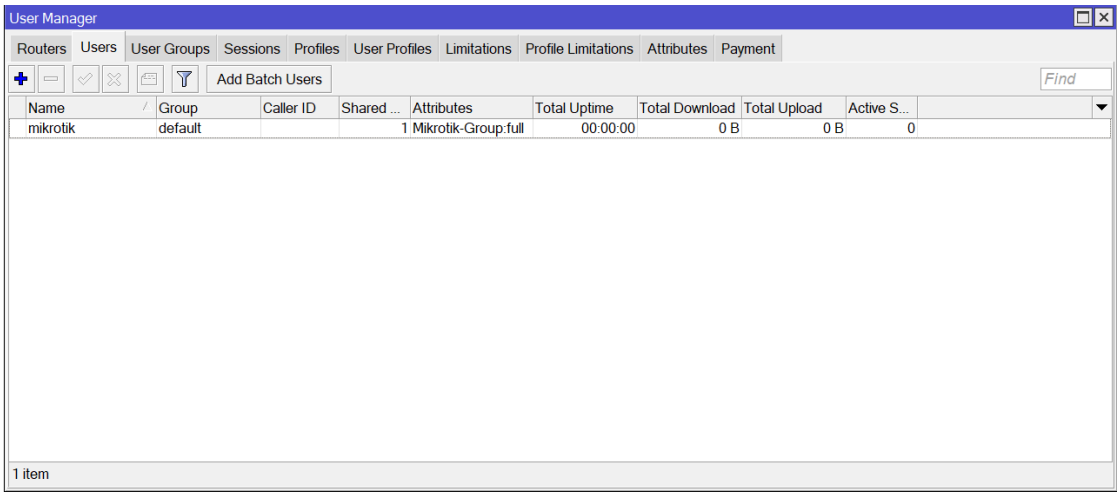
JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885



Gambar 4. Hasil konfigurasi *service User Manager Routers*

d. Menambahkan akun pengguna mikrotik ke User Manager dengan:

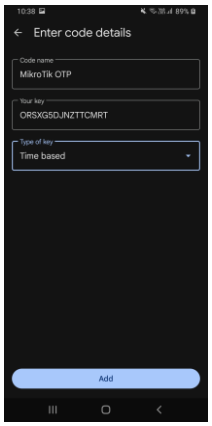
- Grup akses full
- Password admin
- Secret OTP → digunakan untuk menghasilkan kode MFA (OTP) via aplikasi Google Authenticator.



Gambar 5. Hasil konfigurasi *service User Manager Users*

Konfigurasi dari *Gambar 3,* ***Error! Reference source not found.****, Gambar 5, dan Gambar 6* memastikan bahwa setiap akses SSH akan melalui proses autentikasi username, password, dan OTP yang valid dari aplikasi Google Authenticator.

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885



Gambar 6. Konfigurasi Google *Authenticator*

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885

Gambar 7 menunjukkan konfigurasi Google Authenticator dengan menambahkan Secret OTP yang telah dikonfigurasi pada Gambar 6, yang digunakan sebagai faktor kedua dalam penerapan Multi-Factor Authentication (MFA). Pada tahap ini, setiap akses SSH ke MikroTik tidak hanya menggunakan kombinasi username dan password, tetapi juga memerlukan kode OTP yang dihasilkan secara dinamis oleh aplikasi authenticator. Kode OTP tersebut terus berubah dalam interval waktu tertentu sehingga membuat serangan brute force tidak dapat menebak kombinasi login secara konvensional.

2) Konfigurasi Ubuntu 22.04 LTS sebagai mesin penyerang.

TABEL I
KONFIGURASI MESIN UBUNTU

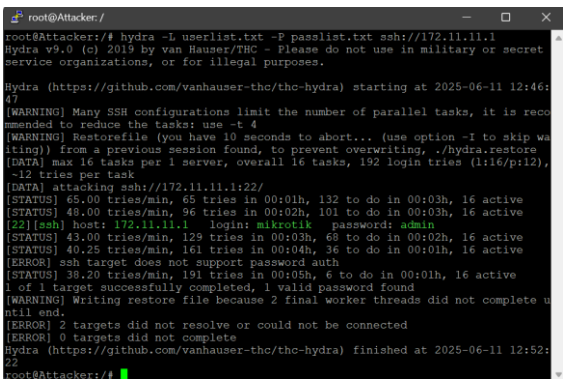
Perintah	Fungsi
apt-get update && apt-get -y install git hydra-gtk default-libmysqlclient-dev libgpg-error-dev libgcrypt-dev libpcre3-dev libidn11-dev libssh-dev libssl-dev make curl gcc	Perbarui indeks paket lalu install paket yang diperlukan.
git clone https://github.com/vanhauser-thc/thc-hydra.git	Unduh salinan kode sumber THC-Hydra dari GitHub ke folder thc-hydra.
cd thc-hydra	Masuk ke direktori kode sumber untuk proses build.
make clean	Hapus hasil build sebelumnya supaya build baru dimulai dari keadaan bersih.
./configure	Periksa dependensi dan buat Makefile sesuai sistem (persiapan untuk kompilasi).
make	Kompilasi kode sumber menjadi binary.
make install	Menginstal binary/library yang sudah dikompilasi ke lokasi sistem

Konfigurasi pada *TABEL I* memastikan agar mesin Ubuntu 22.04 dapat digunakan sebagai perangkat simulasi serangan. Tool THC-Hydra diinstal dari repositori GitHub dan digunakan untuk melakukan serangan brute force terhadap port SSH MikroTik. Konfigurasi ini mencerminkan skenario penyerang dunia nyata yang mencoba mendapatkan akses tidak sah menggunakan teknik dictionary attack.

B. Hasil Pengujian

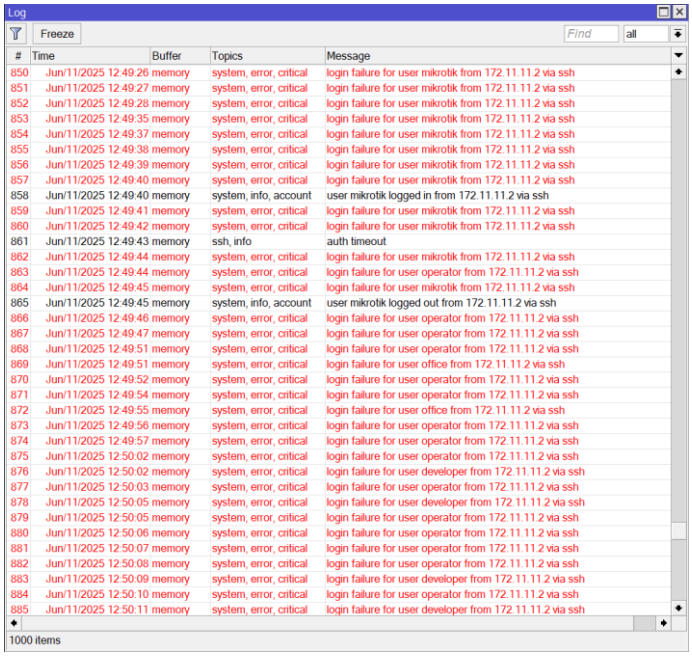
Tabel II
Perbandingan Sistem sebelum dan sesudah mengimplementasikan MFA

	Percobaan	Waktu (Detik)	Status *Brute Force Attack*	*Max CPU Usage*	*Average CPU Usage*	*Max Bandwith (Kb)*	*Average Bandwidth (Kb)*
Tanpa MFA	192	335	Berhasil	14%	2%	9.41Kb	9.41Kb
Dengan MFA	**192**	**332**	**Gagal**	**14%**	**2%**	**9.59Kb**	**9.59Kb**

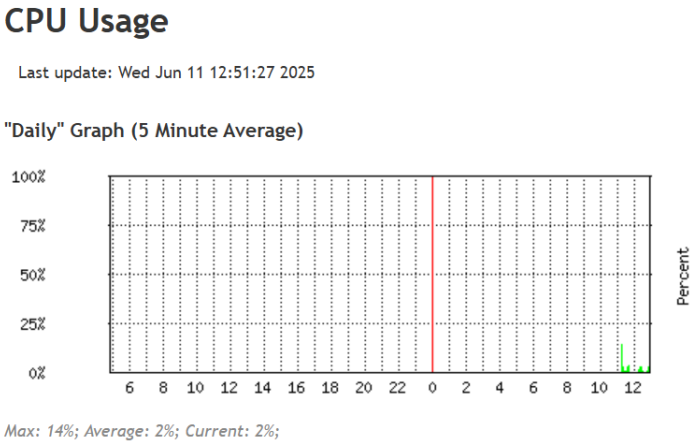


Gambar 7. Operasi *Brute Force* sebelum mengimplementasikan MFA

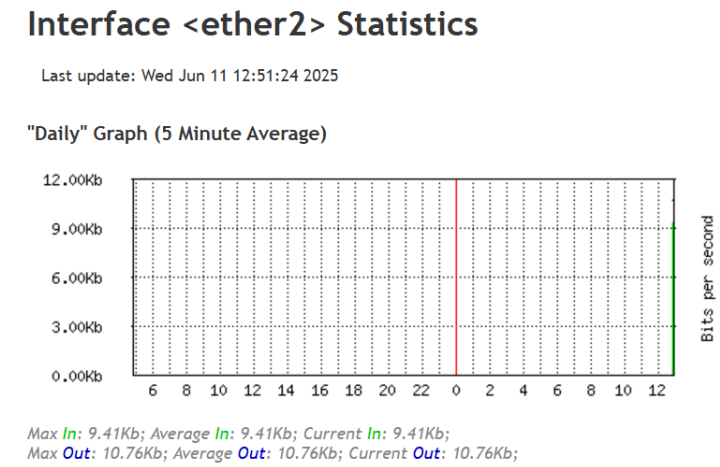
JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885



Gambar 8. *User* mikrotik sebelum mengimplementasikan MFA

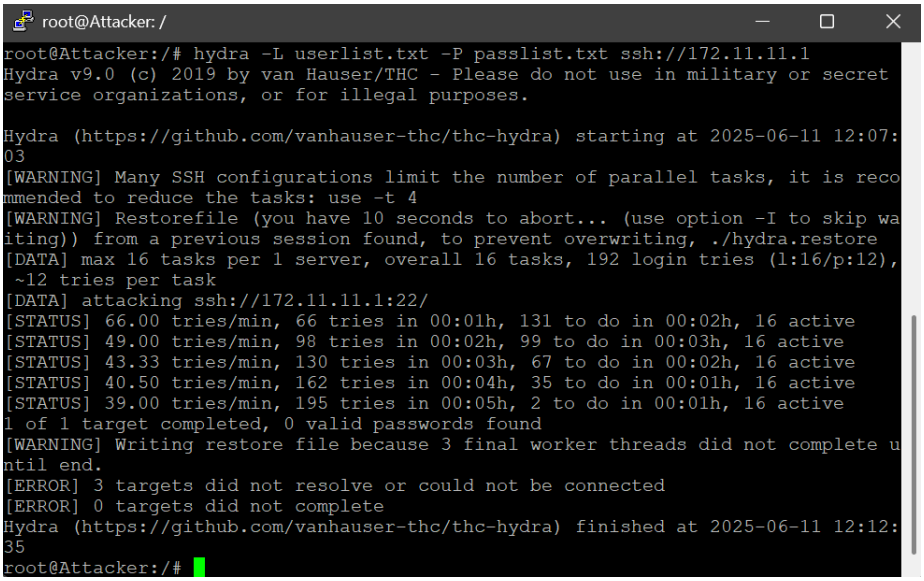


Gambar 9. Grafik *CPU Usage* sebelum mengimplementasikan MFA

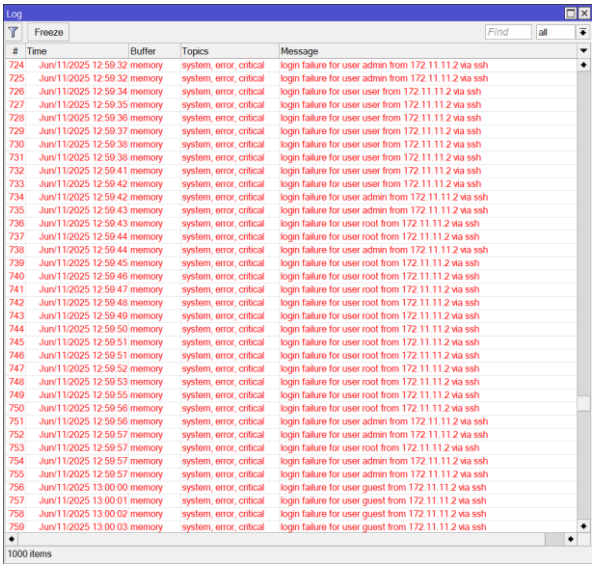


Gambar 10. Grafik *Bandwidth* sebelum mengimplementasikan MFA

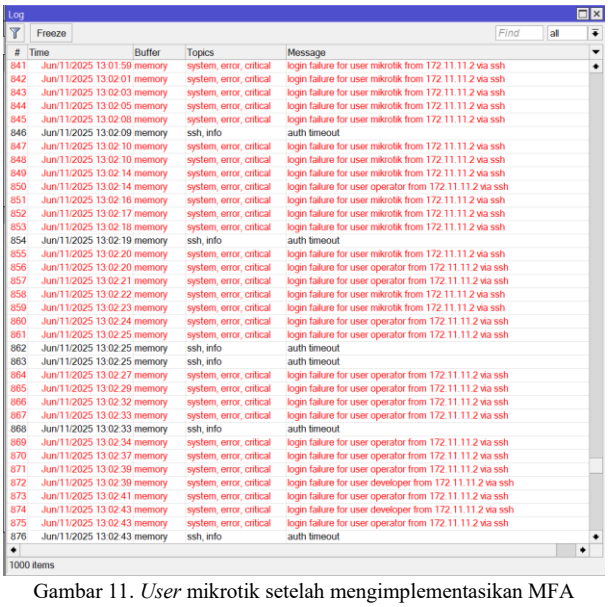
JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885



Gambar 13. Operasi *Brute Force* setelah mengimplementasikan MFA

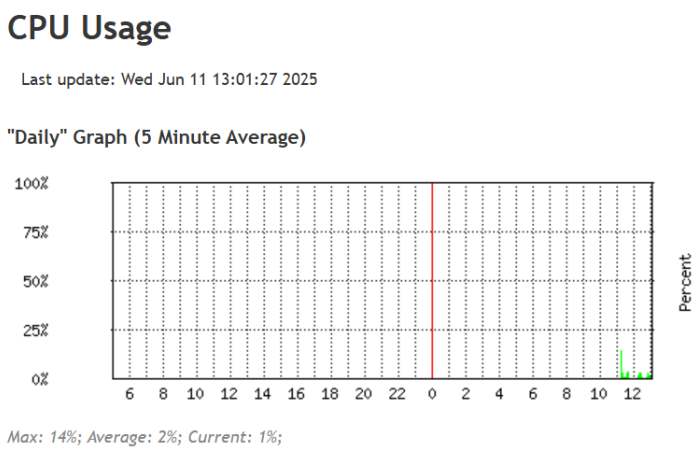


Gambar 12. *Log* MikroTik setelah mengimplementasikan MFA

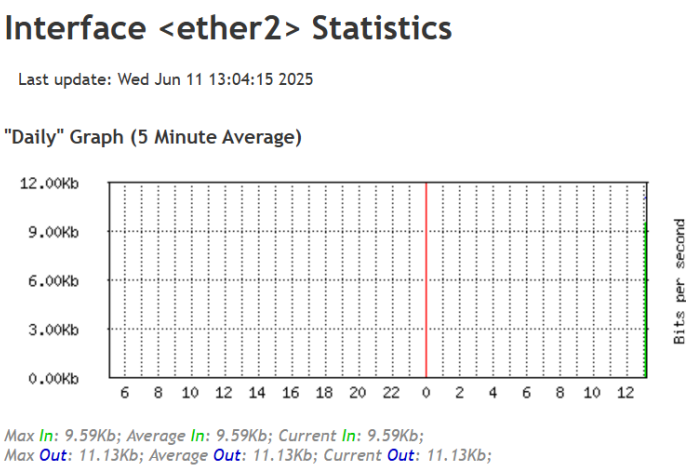


Gambar 11. *User* mikrotik setelah mengimplementasikan MFA

JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)
Journal homepage: https://jurnal.stkippgritulungagung.ac.id/index.php/jipi
ISSN: 2540-8984
Vol. 10, No. 4, December 2025, Pp. 3872-3885



Gambar 15. Grafik *CPU Usage* setelah mengimplementasikan MFA



Gambar 14. Grafik *Bandwidth* setelah mengimplementasikan MFA

C.Pembahasan

Hasil pengujian pada *Tabel II* menunjukkan MFA secara signifikan mengubah status serangan brute force. Pada *Gambar 8*, skenario serangan *Brute Force* dilakukan tanpa MFA, menunjukkan penyerang berhasil menembus sistem SSH MikroTik (akses berhasil) dengan durasi percobaan sekitar 335 detik, sedangkan penerapan MFA membuat serangan gagal sepenuhnya seperti yang terlihat pada *Gambar 13*, dalam durasi hampir sama (sekitar 332 detik). Penambahan lapisan autentikasi berupa OTP dinamis membuat kombinasi kata sandi–kode tidak bisa ditebak penyerang, sehingga brute force konvensional tidak lagi efektif. Temuan ini konsisten dengan laporan bahwa pengguna yang mengaktifkan MFA 99,9% lebih kecil kemungkinannya diterobos serangan otomatis [24]. OTP yang terus berganti pun menambahkan proteksi ekstra, sehingga "membuat [*login*] jauh lebih sulit" bagi penyerang walaupun password diketahui [25]. Dengan demikian, MFA terbukti mengubah hasil serangan dari berhasil menjadi gagal. Temuan ini konsisten dengan literatur mitigasi *brute force* lain. Misalnya, penggunaan sistem pencegahan intrusi (IPS) seperti *Fail2Ban* yang memblokir alamat IP setelah sejumlah kegagalan login [26], atau firewall adaptif lainnya dapat memperlambat dan menekan serangan otomatis. Teknik lain seperti *port knock-ing* juga diusulkan, *port* SSH hanya terbuka setelah urutan "ketukan" rahasia berhasil, sehingga layanan SSH tersembunyi di balik *firewall* hingga autentikasi tambahan [27].

Strategi serangan *brute force* yang diterapkan oleh alat THC-Hydra melibatkan pengujian seluruh kombinasi nama pengguna dan kata sandi dari file kamus yang digunakan. Pada percobaan ini, Hydra menggunakan dua file kamus, userlist.txt berisi 16 nama pengguna dan passlist.txt berisi 12 kata sandi, sehingga total kombinasi yang dicoba adalah 16 × 12 = 192 kombinasi. Pada skenario tanpa MFA, salah satu kombinasi kamus cocok dengan kredensial yang benar. Sebaliknya, pada skenario dengan MFA, meskipun Hydra mungkin mencoba kombinasi user/password yang benar, proses login tetap gagal karena tidak disertai kode OTP yang valid. Oleh karena itu, program Hydra tetap menuntaskan pengujian semua kombinasi nama pengguna dan kata sandi. Meskipun RouterOS membatasi tiga kali upaya login per koneksi SSH [28], penyerang dapat membuka sesi baru secara

berulang sehingga total percobaan mencapai ratusan (sekitar 192 kali pada setiap uji). Akibatnya, durasi serangan yang tercatat menjadi hampir sama pada kedua skenario tersebut meskipun hasil akhirnya berbeda. Perlu diperhatikan bahwa kode OTP berbasis waktu (*Time-based One-Time Password* atau TOTP) hanya menghasilkan kode yang valid dalam jangka waktu singkat (sekitar 30 detik) dan memerlukan sinkronisasi waktu antara *server* autentikasi dan perangkat pengguna. Tanpa perangkat atau aplikasi penghasil kode OTP yang sesuai, atau jika kode yang dimasukkan tidak valid dalam interval waktunya, setiap percobaan *login* akan tetap gagal meskipun nama pengguna dan kata sandi sudah benar. Dengan kata lain, tanpa kode OTP yang valid dalam rentang waktunya, proteksi MFA tidak dapat dilewati, sehingga semua upaya login tetap gagal.

Dari sisi performa sistem, penerapan MFA tidak menimbulkan beban yang signifikan. Dapat dilihat pada *Gambar 11*, dan *Gambar 16*, pemantauan penggunaan CPU menunjukkan angka maksimum sekitar 14% dan rata-rata 2% pada kedua skenario. Demikian pula, konsumsi bandwidth hanya berubah tipis (sekitar 9,41 Kb tanpa MFA menjadi 9,59 Kb dengan MFA) seperti yang terlihat pada *Gambar 12*, dan *Gambar 17*. Angka-angka ini menunjukkan bahwa mekanisme verifikasi OTP meski menambah sedikit lalu lintas jaringan tidak mengakibatkan peningkatan pemakaian CPU atau bandwidth yang berarti. Dengan kata lain, MFA meningkatkan lapisan keamanan tanpa mengorbankan kinerja perangkat.

Analisis log dari **Error! Reference source not found.**, *Gambar 10*, *Gambar 14*, dan *Gambar 15*, menguatkan kesimpulan tersebut. Berdasarkan praktik standar, setiap upaya login SSH dicatat oleh sistem (rsyslog) dalam file log [29]. Contoh entri log pada RouterOS misalnya '*login failure for user* [nama] *from* [IP] *via ssh*' untuk setiap upaya autentikasi yang gagal [30]. Dalam percobaan ini, skenario tanpa MFA akan menghasilkan satu entri "*login success*" (akses diterima) setelah password benar, sedangkan skenario MFA menampilkan entri "login failure" berulang karena meski password mungkin valid, proses gagal saat kode OTP tidak sesuai. Dengan demikian, pola log (failure berulang pada lapisan OTP) menjadi indikator jelas bahwa multi-faktor autentikasi bekerja sebagai "gerbang akhir" yang memblokir akses tidak sah.

Analisis risiko residu menunjukkan bahwa meskipun MFA meningkatkan pertahanan, vektor serangan lain masih ada. Literatur keamanan mengungkap teknik bypass MFA modern. Kondracki dkk. mendokumentasikan bahwa *toolkit phising* MITM (*man-in-the-middle*) berfungsi sebagai proxy terbalik yang memantulkan situs target sambil mencuri kredensial pengguna secara *real-time*. Dalam konteks OTP, ini berarti setiap kode autentikasi satu kali (misalnya 2FA/OTP) yang dimasukkan pengguna akan ditangkap oleh penyerang saat transaksi berlangsung. Kondracki dkk. secara eksplisit menggambarkan MITM-*phishing* sebagai *server proxy* yang "*mirrors a target web page to a victim while harvesting credentials, 2FA codes, and web page content in transit*" [31].

Pengembangan masa depan dapat mencakup penggunaan MFA berbasis perangkat keras (*hardware*). Standar *FIDO2/WebAuthn*, misalnya, menggunakan kunci kriptografis yang tersimpan di token fisik (*USB/NFC*) dan memerlukan bukti kehadiran pengguna, sehingga lebih tahan terhadap *phishing*. Penelitian dan praktek industri menunjukkan kunci keamanan (seperti YubiKey) mampu menyimpan private key secara aman dan mengautentikasi tanpa mengirim token ke jaringan, sehingga hampir mustahil disadap dari jarak jauh [32].

Temuan penelitian ini relevan dengan prinsip arsitektur keamanan modern *Zero Trust*. *Zero Trust Architecture* (ZTA) sesuai dokumen resmi NIST SP 800-207 tentang Arsitektur *Zero Trust*, setiap proses autentikasi dan otorisasi harus dilakukan secara eksplisit dan dinamis sebelum akses diizinkan [33]. Artinya, seluruh subjek dan sumber daya dalam sistem harus diverifikasi (autentikasi) dan diperiksa hak aksesnya (otorisasi) setiap kali mengajukan akses, tanpa asumsi kepercayaan implisit sebelum izin diberikan. Penggunaan MFA, terutama implementasi MFA berbasis waktu seperti TOTP, sejalan dengan prinsip *verify* dan *authenticate* dalam ZTA, di mana setiap permintaan akses harus diperiksa kembali menggunakan faktor yang dinamis. Hasil penelitian mendukung penerapan ZTA di lingkungan jaringan dengan menunjukkan bahwa lapisan MFA dapat mencegah kompromi kredensial sekaligus menjadi bagian penting dari sistem keamanan bertahap (*layered security*).

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian brute force menggunakan THC-Hydra pada SSH MikroTik, dapat disimpulkan bahwa penerapan MFA sangat efektif dalam mencegah serangan tersebut. Pada sistem tanpa MFA, serangan berhasil menembus autentikasi dalam waktu sekitar 335 detik, sedangkan pada sistem dengan MFA, serangan tidak berhasil menembus sistem karena terjadi kegagalan login yang terus berulang selama durasi pengujian selama 332 detik. Hasil ini mengonfirmasi bahwa MFA secara signifikan memperkuat keamanan SSH MikroTik. Selain itu, penerapan MFA tidak menimbulkan beban kinerja yang berarti pada sistem. Penggunaan CPU tetap rendah dengan nilai maksimum sebesar 14 persen dan rata-rata sebesar 2 persen, sedangkan konsumsi bandwidth berada pada kisaran yang stabil antara 9,41 hingga 9,59 Kb pada kedua skenario. Dengan demikian, penerapan MFA dapat meningkatkan keamanan sistem tanpa mengganggu performa perangkat jaringan.

Berdasarkan temuan tersebut, disarankan agar sistem MFA diimplementasikan pada lingkungan jaringan produksi untuk memperkuat pertahanan terhadap serangan brute force. Implementasi nyata MFA akan membantu organisasi mengamankan akses SSH secara lebih andal. Selain itu, pengembangan metode autentikasi lanjutan, misalnya berbasis biometrik atau FIDO, perlu dipertimbangkan guna meningkatkan fleksibilitas dan keamanan sistem autentikasi di masa mendatang.

DAFTAR PUSTAKA

[1] H. Haeruddin, S. E. Prasetyo, dan A. W. Kaharuddin, "Optimalisasi Keamanan Jaringan Di Era Digital menggunakan metode Zero Trust," *J. Inf. Syst. Technol.*, vol. 5, no. 3, hlm. 15–24, Des 2024, doi: 10.37253/joint.v5i3.9986.
[2] V. Grover, "An Efficient Brute Force Attack Handling Techniques for Server Virtualization," *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.3564447.
[3] C. Pamungkas, P. Hendradi, D. Sasongko, dan A. Ghifari, "Analysis of Brute Force Attacks Using National Institute Of Standards And Technology (NIST) Methods on Routers," *J. Inform. Inf. Syst. Softw. Eng. Appl. INISTA*, vol. 5, no. 2, hlm. 115–125, Mei 2023, doi: 10.20895/inista.v5i2.1039.
[4] R. P. Aji, "Analisis Log Serangan Bruteforce Terhadap Web Server Nginx Pada Dasbor Sistem Pencatatan Log Teroptimasi Menggunakan Metode Investigasi Forensik," Des 2022, Diakses: 14 November 2025. [Daring]. Tersedia pada: https://dspace.uii.ac.id/handle/123456789/42416
[5] "2025-dbir-data-breach-investigations-report.pdf." Diakses: 14 November 2025. [Daring]. Tersedia pada: https://www.verizon.com/business/resources/Td8e/reports/2025-dbir-data-breach-investigations-report.pdf
[6] S. Sujalwo, "Manajemen Jaringan Komputer Dengan Menggunakan Mikrotik Router (Computer Network Management Used With Microtic Router)," *Komuniti J. Komun. Dan Teknol. Inf.*, vol. 2, no. 2, hlm. 32–43.
[7] Y. Mulyanto dan A. Algi Fari, "ANALISIS KEAMANAN LOGIN ROUTER MIKROTIK DARI SERANGAN BRUTEFORCE MENGGUNAKAN METODE PENETRATION TESTING (Studi Kasus: SMK NEGERI 2 SUMBAWA)," *J. Inform. Teknol. Dan Sains*, vol. 4, no. 3, hlm. 145–155, Agu 2022, doi: 10.51401/jinteks.v4i3.1897.
[8] D. J. Barrett dan R. E. Silverman, *SSH, the secure shell: the definitive guide*, 1st ed. Cambridge [Mass.]: O'Reilly, 2001.
[9] A. Fauzi, F. Firmansyah, dan T. A. A. Sandi, "Perancangan Keamanan Router Mikrotik Dari Serangan FTP Dan SSH Brute Force," *J. Infortech*, vol. 6, no. 1, hlm. 9–14, Jun 2024, doi: 10.31294/infortech.v6i1.21697.
[10] D. X. Song, D. Wagner, dan X. Tian, "Timing Analysis of Keystrokes and Timing Attacks on {SSH}," dipresentasikan pada 10th USENIX Security Symposium (USENIX Security 01), 2001. Diakses: 13 November 2025. [Daring]. Tersedia pada: https://www.usenix.org/conference/10th-usenix-security-symposium/timing-analysis-keystrokes-and-timing-attacks-ssh
[11] R. Andrews, D. A. Hahn, dan A. G. Bardas, "Measuring the Prevalence of the Password Authentication Vulnerability in SSH," dalam *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland: IEEE, Jun 2020, hlm. 1–7. doi: 10.1109/ICC40277.2020.9148912.
[12] F. Bäumer, M. Brinkmann, dan J. Schwenk, "Terrapin Attack: Breaking SSH Channel Integrity By Sequence Number Manipulation," 8 Mei 2024, *arXiv*: arXiv:2312.12422. doi: 10.48550/arXiv.2312.12422.
[13] "System Security Configuration Guide for Cisco NCS 5500 Series Routers, IOS XR Release 24.1.x, 24.2.x, 24.3.x, 24.4.x - Implementing Secure Shell [Cisco Network Convergence System 5500 Series]," Cisco. Diakses: 14 November 2025. [Daring]. Tersedia pada: https://www.cisco.com/c/en/us/td/docs/iosxr/ncs5500/security/24xx/configuration/guide/b-system-security-cg-ncs5500-24xx/implementing-secure-shell.html
[14] H. Haeruddin, S. E. Prasetyo, dan A. Mindy, "Implementasi Multi-Factor Authentication Untuk Optimalisasi Keamanan Akses Data Di PT.ABC," *J. Manaj. Inform. JAMIKA*, vol. 15, no. 1, hlm. 85096, Feb 2025, doi: 10.34010/5rdjmw37.
[15] S. Setyowibowo, S. Sujito, dan N. Moka, "Keamanan Jaringan Hotspot Dengan Simple Port Knocking Dan Automated Backup Menggunakan Mikrotik," *J. Ilm. Komputasi*, vol. 21, no. 4, Des 2022, doi: 10.32409/jikstik.21.4.3109.
[16] A. P. Usman, R. Y. Bakti, dan M. A. Hayat, "Optimalisasi Sistem Keamanan SSH dari Serangan Brute Force Mengunakan Intrusion Prevention System pada Mikrotik," *Arus J. Sains Dan Teknol.*, vol. 2, no. 1, hlm. 116–122, Apr 2024, doi: 10.57250/ajst.v2i1.380.
[17] Tony Sanjaya dan Didik Setiyadi, "Network Development Life Cycle (NDLC) dalam Perancangan Jaringan Komputer pada Rumah Shalom Mahanaim," *J. Mhs. Bina Insani*, vol. 4, no. 1, hlm. 1–10, Agu 2019.
[18] F. Naim, Rd. R. Saedudin, dan U. Y. K. S. Hediyanto, "ANALYSIS OF WIRELESS AND CABLE NETWORK QUALITY-OF-SERVICE PERFORMANCE AT TELKOM UNIVERSITY LANDMARK TOWER USING NETWORK DEVELOPMENT LIFE CYCLE (NDLC) METHOD," *JIPI J. Ilm. Penelit. Dan Pembelajaran Inform.*, vol. 7, no. 4, hlm. 1033–1044, Nov 2022, doi: 10.29100/jipi.v7i4.3192.
[19] Y. Ardiansyah, U. Y. Kurnia Septo Hediyanto, dan M. T. Kurniawan, "ANALISIS DAN OPTIMASI TEKNOLOGI JARINGAN WIRELESS PADA RUANGAN PROSES MANUFAKTUR DI GEDUNG MANGUDU UNIVERSITAS TELKOM DENGAN MENGGUNAKAN WIRELESS SITE SURVEY," *JIPI J. Ilm. Penelit. Dan Pembelajaran Inform.*, vol. 9, no. 2, hlm. 529–539, Mei 2024, doi: 10.29100/jipi.v9i2.4483.
[20] S. Donaldson, N. Coull, dan D. McLuskie, "A methodology for testing virtualisation security," dalam *2017 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA)*, London, United Kingdom: IEEE, Jun 2017, hlm. 1–8. doi: 10.1109/CyberSA.2017.8073397.
[21] C. A. Hamka, H. Sajati, dan Y. Indrianingsih, "SISTEM KEAMANAN JAIL BASH UNTUK MENGAMANKAN AKUN LEGAL DARI KEJAHATAN INTERNET MENGGUNAKAN THC-HYDRA," *Compiler*, vol. 3, no. 1, Mei 2014, doi: 10.28989/compiler.v3i1.63.
[22] Yopi Hidayatul Akbar, "Evaluasi Keamanan Jaringan Wireless Hotspot Menggunakan Metode Square (Studi Kasus Warnet Medianet Sumedang)," *Infoman's*, vol. 9, no. 2, hlm. 75–90, 2015, doi: 10.33481/infomans.v9i2.60.
[23] D. M'Raihi, J. Rydell, M. Pei, dan S. Machani, "TOTP: Time-Based One-Time Password Algorithm," Internet Engineering Task Force, Request for Comments RFC 6238, Mei 2011. doi: 10.17487/RFC6238.
[24] L. Meyer, S. Romero, G. Bertoli, T. Burt, A. Weinert, dan J. Lavista Ferres, *How effective is multifactor authentication at deterring cyberattacks?* 2023. doi: 10.48550/arXiv.2305.00945.
[25] "How to Stop Brute Force Attacks with WordPress OTP," Shield Security. Diakses: 18 Juni 2025. [Daring]. Tersedia pada: https://getshieldsecurity.com/blog/wordpress-one-time-password/
[26] R. George dan E. Z. Abay, *Detection of SSH Brute-Force Attacks Using Machine Learning : A Comparative Study with Fail2Ban and PAM Tally2*. 2025. Diakses: 13 November 2025. [Daring]. Tersedia pada: https://urn.kb.se/resolve?urn=urn:nbn:se:hv:diva-23725
[27] D. Patel, D. Trivedi, U. Raval, dan A. Dennisan, "2F-Authsys: A hyperlocal two-factor authentication system using Near Sound Data Transfer," *J. Appl. Res. Technol.*, vol. 22, no. 2, hlm. 197–205, Apr 2024, doi: 10.22201/icat.24486736e.2024.22.2.2244.
[28] "Bruteforce prevention - RouterOS - MikroTik Documentation." Diakses: 18 Juni 2025. [Daring]. Tersedia pada: https://help.mikrotik.com/docs/spaces/ROS/pages/268337176/Bruteforce+prevention
[29] M. Cezar, "How to Find All Failed SSH login Attempts in Linux." Diakses: 18 Juni 2025. [Daring]. Tersedia pada: https://www.tecmint.com/find-failed-ssh-login-attempts-in-linux/
[30] "Login failure on log - RouterOS / Beginner Basics," MikroTik community forum. Diakses: 18 Juni 2025. [Daring]. Tersedia pada: https://forum.mikrotik.com/t/login-failure-on-log/111336
[31] B. Kondracki, B. A. Azad, O. Starov, dan N. Nikiforakis, "Catching Transparent Phish: Analyzing and Detecting MITM Phishing Toolkits," dalam *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, dalam CCS '21. New York, NY, USA: Association for Computing Machinery, Nov 2021, hlm. 36–50. doi: 10.1145/3460120.3484765.

[32] R. Soni, "Understanding OTP Authentication: HOTP & TOTP Explained," LoginRadius. Diakses: 14 November 2025. [Daring]. Tersedia pada: https://www.loginradius.com/blog/identity/what-is-otp-authentication
[33] S. Rose, O. Borchert, S. Mitchell, dan S. Connelly, "Zero Trust Architecture," National Institute of Standards and Technology, Agu 2020. doi: 10.6028/NIST.SP.800-207.