# Meningkatkan Keamanan Autentikasi pada Web Statis dengan QR Code, OAuth2, dan STP

**Muhammad Rifqi Daffa Ulhaq1, Rolly Maulana Awangga2, Yusril Helmi Setyawan3**

1Universitas Logistik dan Bisnis Internasional, Indonesia

1Universitas Logistik dan Bisnis Internasional, Indonesia

1Universitas Logistik dan Bisnis Internasional, Indonesia

Email: 1204045@std.ulbi.ac.id

*Received* 30 November 201x**|** *Revised* 30 Desember 201x**|** *Accepted* 30 Januari 201x

## Abstrak

Keamanan dalam autentikasi pengguna adalah aspek krusial dalam pengembangan sistem web, terutama web statis yang sering kali kurang aman dibandingkan web dinamis. Penelitian ini mengembangkan sistem autentikasi yang menggabungkan QR code, OAuth2 (Google), dan short temporary password (STP) untuk meningkatkan keamanan web statis. Kombinasi ini diharapkan memberikan solusi autentikasi berlapis yang lebih kuat dalam mencegah akses tidak sah, dengan QR code memanfaatkan perangkat mobile, OAuth2 menawarkan autentikasi token tanpa menyimpan kata sandi, dan STP memberikan keamanan tambahan dengan kata sandi sementara. Selain itu, teknik CAPTCHA dan rate limiting diterapkan untuk mencegah serangan otomatis dan memantau permintaan mencurigakan. Evaluasi sistem melibatkan pengujian brute force dan dictionary attack masing-masing selama 5 menit, menunjukkan ketidakberhasilan serangan ini dalam menebak kata sandi. Implementasi dan evaluasi menunjukkan peningkatan signifikan dalam keamanan autentikasi web statis, mengatasi kelemahan pendekatan konvensional dan menawarkan pendekatan yang dapat diadopsi pengembang web statis untuk meningkatkan perlindungan data dan pengguna.

**Kata kunci**: Keamanan Web, Autentikasi, Otorisasi, Web Statis.

## Abstract

*Security in user authentication is a crucial aspect in web system development, especially static web which is often less secure than dynamic web. This research develops an authentication system that combines QR code, OAuth2 (Google), and short temporary password (STP) to improve static web security. This combination is expected to provide a layered authentication solution that is stronger in preventing unauthorized access, with QR codes utilizing mobile devices, OAuth2 offering token authentication without storing passwords, and STP providing additional security with temporary passwords. In addition, CAPTCHA and rate limiting techniques are implemented to prevent automated attacks and monitor suspicious requests. System evaluation involved brute force and dictionary attack testing for 5 minutes each, showing the unsuccessfulness of these attacks in guessing passwords. The implementation and evaluation show significant improvements in static web authentication security, addressing the weaknesses of conventional approaches and offering an approach that static web developers can adopt to improve data and user protection.*

***Keywords****: Web Security, Authentication, Authorization, Static Web.*

## 1. PENDAHULUAN

Keamanan merupakan aspek fundamental dalam pengembangan sistem web, khususnya pada era digital saat ini di mana ancaman terhadap data dan informasi semakin meningkat[1]. Menurut laporan dari Cybersecurity Ventures, kerugian global akibat kejahatan siber diperkirakan akan mencapai $10,5 triliun per tahun pada 2025[2]​​. Peningkatan signifikan ini menekankan urgensi untuk mengembangkan sistem keamanan yang lebih efektif dan efisien dalam melindungi data pengguna. Autentikasi pengguna adalah salah satu elemen kunci dalam menjaga keamanan sistem, dengan tujuan untuk memastikan bahwa hanya pengguna yang sah yang dapat mengakses sumber daya yang dilindungi[3]. Web statis, yang sering kali dianggap lebih sederhana dan kurang dinamis dibandingkan dengan web dinamis, tetap menghadapi tantangan keamanan yang signifikan[4]. Oleh karena itu, pengembangan metode autentikasi yang efektif dan aman menjadi sangat penting.

Berbagai metode autentikasi telah dikembangkan dan diterapkan dalam upaya meningkatkan keamanan sistem web. Salah satu metode yang populer adalah penggunaan QR code, yang menawarkan kemudahan dan kenyamanan bagi pengguna dengan memanfaatkan perangkat mobile untuk proses autentikasi. Penelitian[5] menunjukkan bahwa penggunaan QR code dapat menggantikan kebutuhan akan tabel verifikasi kata sandi dan memanfaatkan ponsel sebagai token yang praktis dan ekonomis. Selain itu, metode Time-Based One-Time Password (TOTP) yang dikombinasikan dengan QR code telah terbukti meningkatkan keamanan sistem gateway[6], [7].

Metode autentikasi lain yang semakin populer adalah OAuth2, yang menyediakan mekanisme autentikasi berbasis token yang aman tanpa perlu menyimpan kata sandi. OAuth2 telah diadopsi secara luas oleh berbagai layanan web untuk memastikan keamanan akses data dan layanan. Penelitian[8], [9] membahas mekanisme dan keamanan OAuth2 dalam konteks layanan web dan aplikasi mobile.

Dalam konteks short temporary password (STP), metode ini menawarkan keamanan tambahan dengan memberikan kata sandi sementara yang hanya berlaku untuk satu sesi autentikasi. Penelitian[10], [11] mengusulkan sistem autentikasi yang menggunakan QR code untuk lingkungan cloud computing mobile, sementara[12] membahas sistem autentikasi berbasis QR code kontekstual yang menyimpan informasi berubah-ubah untuk setiap sesi autentikasi. Dalam penelitian ini, STP dikirimkan kepada pengguna melalui WhatsApp agar mudah diadopsi, sehingga dapat meningkatkan pengalaman pengguna karena WhatsApp sangat populer dan digandrungi belakangan ini[13].

Meskipun berbagai metode autentikasi telah dikembangkan, penelitian yang menggabungkan ketiga metode ini QR code, OAuth2, dan STP untuk web statis masih belum banyak ditemukan. Oleh karena itu, penelitian ini bertujuan untuk mengisi kekosongan tersebut dengan mengembangkan sistem autentikasi yang menggabungkan ketiga metode tersebut, memberikan solusi autentikasi yang lebih kuat dan berlapis untuk web statis. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan keamanan web statis dan memberikan pendekatan autentikasi yang inovatif dan efektif bagi pengembang web.

## 2. METODE PENELITIAN

Penelitian ini mengembangkan sistem autentikasi untuk web statis dengan menggabungkan tiga metode autentikasi yang berbeda, yaitu QR code, OAuth2 (Google), dan short temporary password (STP). Bagian ini menjelaskan secara rinci tahapan-tahapan yang dilakukan dalam pengembangan dan implementasi sistem tersebut.

### 2.1 Desain Sistem

Sistem autentikasi yang diusulkan terdiri dari tiga komponen utama:

1. **Autentikasi QR Code**

Pengguna akan memindai QR code yang ditampilkan pada halaman login web menggunakan perangkat mobile mereka.

1. **OAuth2 (Google)**

Pengguna akan dapat login menggunakan akun Google mereka, yang menggunakan protokol OAuth2 untuk memberikan token autentikasi yang aman.

1. **Secure Temporary Password (STP)**

Sistem akan mengirimkan STP ke pengguna melalui WhatsApp untuk memastikan keamanan tambahan. STP ini hanya berlaku untuk satu sesi dan memiliki masa berlaku yang singkat.

### 2.2 Implementasi Sistem

1. **Autentikasi QR Code**

Pengguna akan memindai QR code yang ditampilkan pada halaman login web menggunakan perangkat mobile mereka. Untuk merinci proses autentikasi ini, Berikut adalah alur dari proses autentikasi menggunakan QR code.

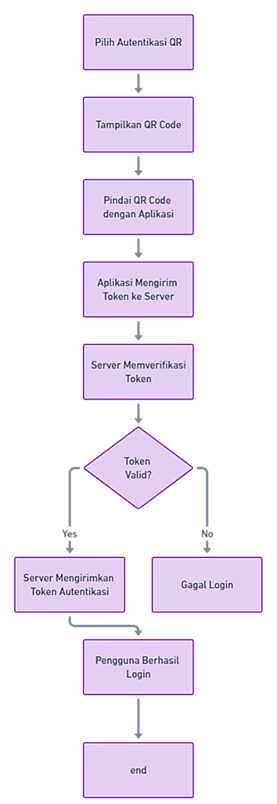
* Pengguna memilih untuk mengautentikasi menggunakan kode QR.
* Aplikasi menampilkan kode QR untuk dipindai oleh pengguna.
* Pengguna memindai kode QR menggunakan kamera dari aplikasi Whatsapp.
* Aplikasi mobile mengirimkan token yang dipindai ke server.
* Server memeriksa validitas token.
* IF token is valid THEN

Server mengirim token Autentikasi kepada pengguna.

Pengguna berhasil login.

ELSE

Login gagal.



**Gambar 1 Autentikasi QR Code**

1. **OAuth2 (Google)**

Pengguna akan dapat login menggunakan akun Google mereka, yang menggunakan protokol OAuth2 untuk memberikan token autentikasi yang aman. Untuk merinci proses autentikasi ini, Berikut adalah alur dari proses autentikasi menggunakan google.

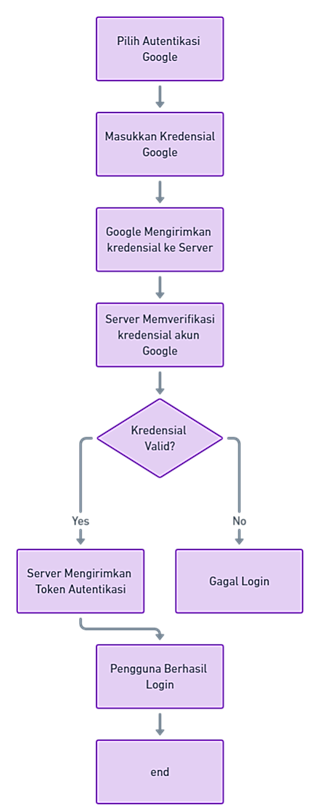
* Pengguna memilih untuk mengautentikasi menggunakan Google.
* Pengguna mengklik tombol "Login with Google".
* Pengguna memasukkan kredensial Google mereka.
* Google mengirimkan kredensial ke server.
* Server memverifikasi kredensial Google yang diterima.
* Server memeriksa validitas kredensial dari akun Google.
* IF crendential is valid THEN

Server mengirim token Autentikasi kepada pengguna.

Pengguna berhasil login.

ELSE

Login gagal.



**Gambar 2 Autentikasi OAuth2 (Google)**

1. **Secure Temporary Password (STP)**

Sistem akan mengirimkan STP ke pengguna melalui WhatsApp untuk memastikan keamanan tambahan. STP ini hanya berlaku untuk satu sesi dan memiliki masa berlaku yang singkat. Untuk merinci proses autentikasi ini, Berikut adalah alur dari proses autentikasi menggunakan STP.

* Pengguna memilih untuk mengautentikasi menggunakan STP.
* Pengguna memasukkan nomor telepon mereka.
* Sistem memverifikasi nomor telepon yang diberikan.
* Sistem memeriksa validitas nomor telepon.
* IF phonenumber is valid THEN

Server membuat password.

ELSE

Login gagal.

* Sistem menyimpan password yang di-hash dengan waktu kedaluwarsa 4 menit.
* Sistem mengirimkan password ke pengguna melalui WhatsApp.
* Pengguna menerima password melalui WhatsApp.
* Pengguna memasukkan password yang diterima pada halaman login.
* Sistem memverifikasi password yang dimasukkan dan memeriksa apakah sudah kedaluwarsa.
* Sistem memeriksa apakah password valid dan belum kedaluwarsa.
* IF password is valid THEN

Server mengirimkan token autentikasi kepada pengguna.

Pengguna berhasil login.

ELSE IF password is expired THEN

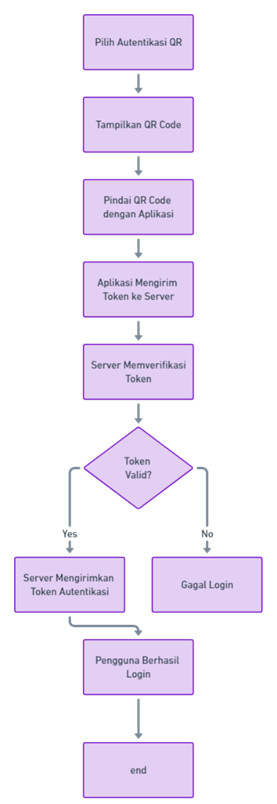
Pengguna mengklik "Resend Password" untuk menerima password baru.

Server mengirimkan token autentikasi kepada pengguna.

Pengguna berhasil login.

ELSE

Login gagal.



**Gambar 3 Autentikasi STP**

## 3. HASIL DAN PEMBAHASAN

**3.1 Hasil**

Penelitian ini menghasilkan skema autentikasi dan otorisasi yang aman pada website statis menggunakan berbagai metode autentikasi, seperti autentikasi QR, autentikasi Google, dan autentikasi STP. Hasil penelitian ini menunjukkan keberhasilan dalam mencapai tujuan tersebut melalui pengujian yang dilakukan.

1. **Autentikasi QR Code**

Autentikasi QR memungkinkan pengguna untuk memindai kode QR yang unik untuk mendapatkan akses ke sistem[14]. Hasil pengujian menunjukkan bahwa metode ini sangat efektif dalam mengamankan akses pengguna, karena setiap kode QR hanya dapat digunakan sekali dan memiliki masa berlaku yang terbatas. Namun, kelemahan dari metode ini adalah pengguna membutuhkan perangkat tambahan yang mendukung pemindaian QR code.

1. **OAuth2**

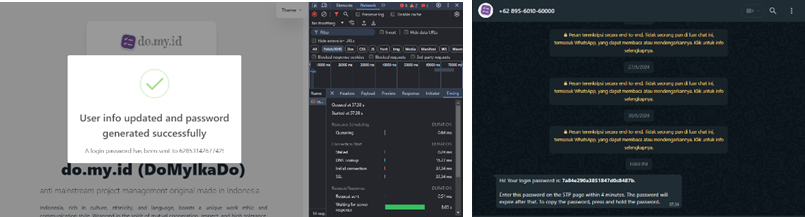
Autentikasi Google menggunakan layanan Google Sign-In untuk memverifikasi identitas pengguna[15]. Hasil pengujian menunjukkan bahwa metode ini mudah digunakan dan sangat aman. Pengguna cukup login menggunakan akun Google mereka, dan sistem memanfaatkan infrastruktur keamanan Google yang handal untuk mengamankan autentikasi pengguna. Kelebihan utama dari metode ini adalah pengguna tidak perlu mengingat password tambahan, karena mereka menggunakan kredensial Google yang sudah ada

1. **Secure** **Temporary** **Password (STP)**

Autentikasi STP (Secure Temporary Password) mengirimkan password sementara melalui WhatsApp ke nomor ponsel pengguna. Hasil pengujian menunjukkan bahwa metode ini efektif dalam mengamankan akses pengguna, terutama bagi mereka yang sering lupa password. Berikut adalah skenario pengujian yang dilakukan untuk metode autentikasi STP:

* Skenario: Berhasil

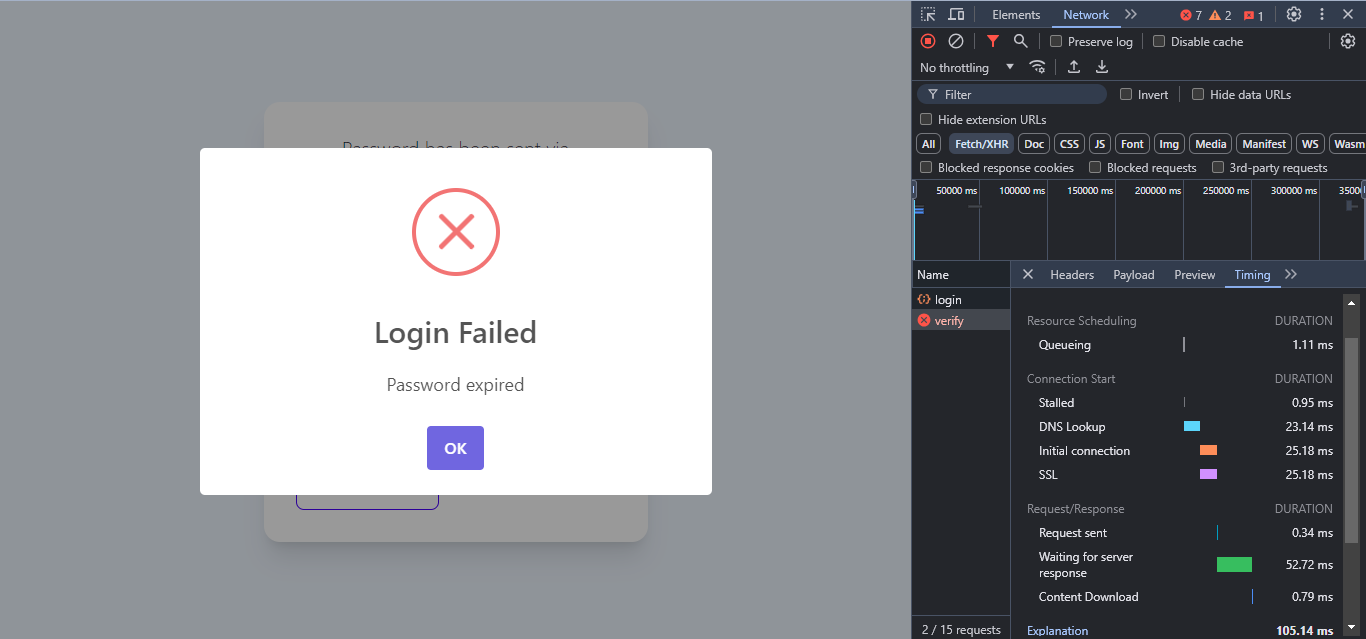
Seperti yang terlihat pada gambar dibawah, pengguna yang memasukkan password dengan benar dalam batas waktu yang ditentukan berhasil masuk ke system.



**Gambar 4 Autentikasi STP: Berhasil**

* Skenario: Expired Password

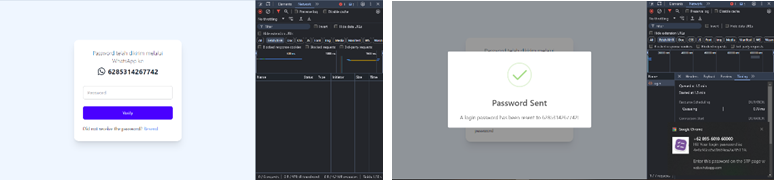
Jika password telah kedaluwarsa, pengguna akan menerima pesan bahwa password telah kedaluwarsa dan harus meminta password baru.



**Gambar 5 Autentikasi STP: Expired Password**

* Skenario: Resend Password

Pengguna dapat mengirim ulang password jika tidak menerima atau jika password telah kedaluwarsa. Metode ini memastikan pengguna tetap dapat mengakses sistem meskipun terjadi kendala pada pengiriman password sebelumnya.

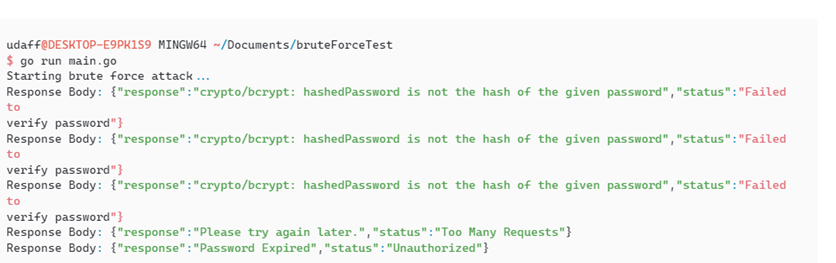


**Gambar 6 Autentikasi STP: Resend Password**

Hasil pengujian menunjukkan bahwa metode autentikasi STP meningkatkan keamanan sistem dengan memanfaatkan faktor autentikasi tambahan seperti pengiriman password melalui WhatsApp.

1. **Hasil Pengujian Brute Force**

Teknik brute force adalah metode serangan di mana penyerang mencoba semua kemungkinan kombinasi untuk menebak kata sandi pengguna[16]. Pengujian penetrasi menggunakan Teknik brute force dilakukan dalam kurun waktu 5 menit. Karena pada tenggat waktu itulah satu password akan berlaku untuk sebuah akun yang tengah melakukan autentikasi. Dalam kurun waktu tersebut akan ditebak berbagai kemungkinan password yang digunakan pengguna untuk melakukan autentikasi. Berikut adalah hasil pengujian menunjukkan serangan brute force yang dilakukan selama 5 menit.



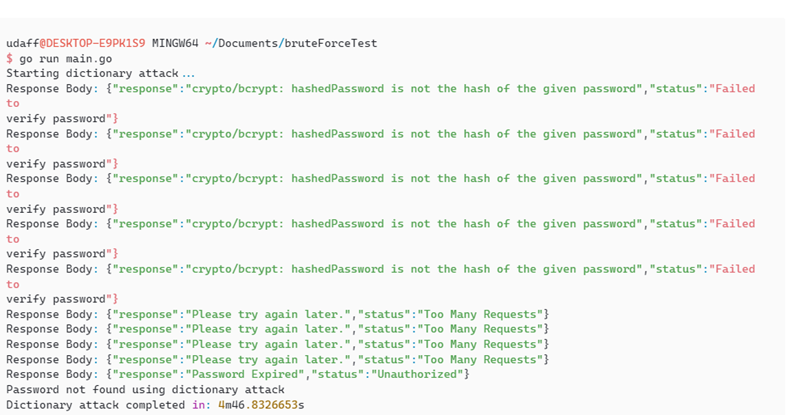
**Gambar 7 Hasil Pengujian Brute Force**

Hasil dari pengujian brute force menunjukkan bahwa metode ini tidak berhasil menemukan password yang valid dalam waktu 5 menit. Selama pengujian, program brute force mencoba berbagai kombinasi password, dan respon dari server menunjukkan beberapa pesan kesalahan seperti "crypto/bcrypt: hashedPassword is not the hash of the given password" dan "Password Expired". Selain itu, mekanisme rate limiting yang diterapkan pada server mengakibatkan beberapa permintaan mendapatkan respon "Too Many Requests", yang berarti server berhasil membatasi jumlah permintaan yang bisa diterima dalam jangka waktu tertentu.

Penting untuk dicatat bahwa tidak semua baris respon dari server dimasukkan dalam hasil ini, namun dapat disimpulkan bahwa selama periode pengujian, brute force terus mencoba berbagai kemungkinan password untuk menebak kata sandi pengguna saat autentikasi, namun tidak ada yang berhasil

1. **Hasil Pengujian Dictionary Attack**

Selain itu, teknik dictionary attack juga digunakan, di mana penyerang menggunakan daftar kata sandi umum untuk menebak kata sandi pengguna[17]. Pengujian dengan teknik dictionary attack dilakukan untuk mengevaluasi keamanan sistem STP (Secure Temporary Password) yang telah dibuat. Teknik dictionary attack adalah metode serangan di mana penyerang menggunakan daftar kata-kata atau kombinasi yang umum digunakan sebagai kata sandi untuk menebak kata sandi pengguna. Dalam penelitian ini, pengujian dictionary attack akan dijalankan selama 5 menit, di mana program akan mencoba mencocokkan setiap kata dalam daftar (<https://github.com/danielmiessler/SecLists/blob/master/Passwords/2023-200_most_used_passwords.txt> ) dengan kata sandi pengguna yang sebenarnya. Tujuan dari pengujian ini adalah untuk menilai seberapa baik sistem dapat bertahan terhadap serangan yang menggunakan kata-kata umum atau yang sering digunakan sebagai kata sandi.



**Gambar 8 Hasil Pengujian Dictionary Attack**

Hasil dari pengujian dictionary attack menunjukkan bahwa teknik ini tidak berhasil menebak kata sandi pengguna dalam waktu 5 menit. Pengujian ini dimulai dengan mencoba berbagai kata sandi yang umum digunakan, yang diambil dari daftar kata yang disimpan dalam file dictionary.txt. Selama pengujian, program menerima beberapa respon dari server, termasuk pesan kesalahan seperti "crypto/bcrypt: hashedPassword is not the hash of the given password," yang menunjukkan bahwa kata sandi yang dicoba tidak cocok dengan hash kata sandi yang disimpan. Selain itu, server juga mengirimkan pesan "Please try again later." yang menunjukkan bahwa batas permintaan telah tercapai, serta pesan "Password Expired" yang menunjukkan bahwa kata sandi sementara yang digunakan telah kedaluwarsa.

Secara keseluruhan, meskipun program pengujian mencoba menebak berbagai kemungkinan password yang digunakan oleh pengguna yang tengah melakukan autentikasi, baik dengan kombinasi karakter maupun daftar kata selama hampir 5 menit, teknik brute force maupun dictionary attack ini tidak berhasil menebak kata sandi pengguna yang sebenarnya.

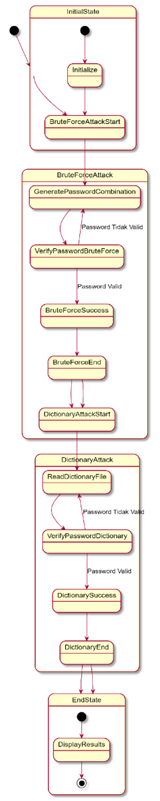
**3.2 Pembahasan**

Penelitian ini membahas tiga metode autentikasi yang diterapkan pada website statis dan mengevaluasi keefektifan masing-masing metode.

1. **Evaluasi Keamanan**

Pada bagian ini, akan di evaluasi keamanan sistem dengan menggunakan metode penetration test. Teknik brute force akan digunakan sebagai salah satu alat utama untuk menguji ketahanan sistem terhadap upaya pembobolan yang agresif. Penetration test ini dirancang untuk mensimulasikan serangan yang mungkin dilakukan oleh pihak tidak bertanggung jawab, dengan tujuan untuk menemukan dan memperbaiki kerentanan yang ada. Hasil dari pengujian ini akan memberikan gambaran yang jelas mengenai efektivitas dan kekuatan sistem autentikasi dalam menghadapi ancaman keamanan yang nyata.

Dalam penelitian ini, dua teknik pegujian penetrasi digunakan untuk menguji keamanan sistem STP (Secure Temporary Password) yang telah dibuat. Logika dari program pengujian penetrasi yang digunakan dalam penelitian ini dapat dijelaskan sebagai berikut:



**Gambar 9 Pengujian Brute Force dan Dictionary Attack**

Diagram State diatas (Pengujian Brute Force dan Dictionary Attack) menggambarkan alur proses dari program pengujian yang dirancang untuk menguji keamanan sistem otentikasi STP. Proses dimulai dari InitialState, bergerak menuju BruteForceAttackStart, di mana serangan brute force dimulai dengan mencoba setiap kombinasi password hingga panjang maksimum yang ditentukan. Dalam state BruteForceAttack, program menghasilkan kombinasi password pada BruteForceGeneratePassword, yang kemudian diverifikasi pada BruteForceVerifyPassword. Jika password tidak valid, proses kembali ke BruteForceGeneratePassword; jika valid, proses bergerak ke BruteForceSuccess, menandakan password ditemukan, dan berakhir pada BruteForceEnd.

Kemudian berlanjut ke DictionaryAttackStart. Pada state DictionaryAttack, program membaca file dictionary pada DictionaryReadFile dan memverifikasi password pada DictionaryVerifyPassword. Jika password tidak valid, proses kembali ke DictionaryReadFile; jika valid, proses bergerak ke DictionarySuccess, menandakan password ditemukan, dan berakhir pada DictionaryEnd, kemudian berlanjut ke EndState. Dalam EndState, hasil dari serangan brute force dan dictionary attack ditampilkan pada DisplayResults, dan proses berakhir.

Diagram ini mengilustrasikan bagaimana program bekerja secara sistematis untuk menguji keamanan sistem otentikasi, melalui langkah-langkah penghasilan dan verifikasi kombinasi password, untuk menemukan password yang valid dan membantu mengidentifikasi potensi kelemahan dalam sistem otentikasi.

1. **Perbandingan Keamanan**

Perbandingan antara program yang memiliki keamanan lemah dengan program yang memiliki keamanan kuat berdasarkan beberapa aspek utama dari sistem otentikasi. Aspek-aspek tersebut mencakup pengelolaan kata sandi, penggunaan CAPTCHA, perlindungan terhadap serangan brute force, dan penanganan permintaan dari pengguna.

* Pengelolaan Kata Sandi

Pengelolaan kata sandi merupakan aspek penting dalam keamanan aplikasi. Kata sandi yang lemah atau tidak dikelola dengan baik dapat membuka celah bagi penyerang untuk mengakses data sensitif. Oleh karena itu, penting untuk memahami bagaimana program yang lemah dan kuat dalam menangani kata sandi guna memastikan keamanan yang optimal.

Program Lemah:

1. Menggunakan kata sandi dalam bentuk teks biasa tanpa enkripsi.
2. Menyimpan kata sandi dalam database tanpa hashing.
3. Menggunakan kata sandi yang sederhana dan mudah ditebak karena berasal dari input pengguna.



**Gambar 10 Ilustrasi program lemah dalam menangani password**

Program Yang Seharusnya:

1. Menggunakan algoritma hashing yang aman, seperti bcrypt, untuk menyimpan kata sandi.
2. Menghasilkan kata sandi yang kompleks dan panjang secara acak.
3. Memanfaatkan salt dalam proses hashing untuk meningkatkan keamanan.



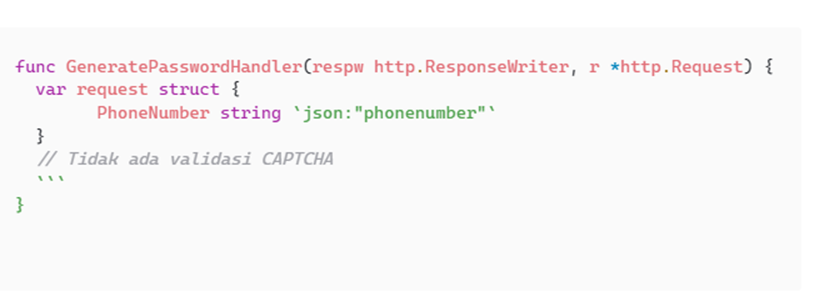
**Gambar 11 Program yang digunakan dalam penelitian untuk menangani password**

* Validasi CAPTCHA

Validasi CAPTCHA adalah langkah penting untuk mencegah serangan otomatis dari bot. CAPTCHA yang efektif dapat membedakan antara pengguna manusia dan bot, sehingga melindungi aplikasi dari serangan brute force dan spam. Bagian ini membahas perbedaan antara program yang lemah dan kuat dalam implementasi CAPTCHA.

Program Lemah:

1. Tidak menggunakan CAPTCHA untuk validasi pengguna.
2. Mengabaikan validasi CAPTCHA pada langkah-langkah kritis, seperti login atau pengiriman kata sandi.



**Gambar 12 Ilustrasi program yang tidak memiliki validasi CAPTCHA**

Program yang seharusnya:

1. Mengintegrasikan CAPTCHA yang kuat, yang dalam hal ini, penelitian ini menggunakan Cloudflare Turnstile, sehingga mengenali pola serangan otomatis dan bot.
2. Memastikan CAPTCHA selalu aktif dan diperiksa pada setiap permintaan otentikasi.

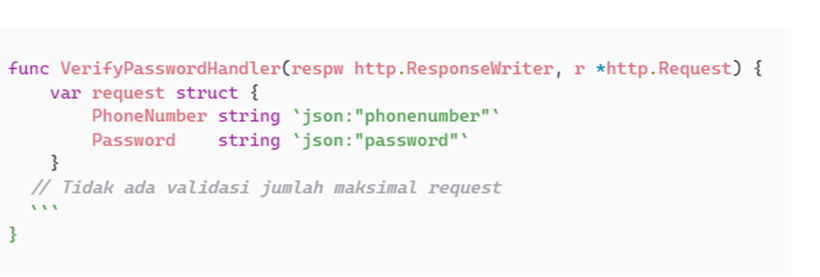


**Gambar 13 Program yang digunakan penelitian untuk validasi CAPTCHA**

* Penanganan Permintaan (Request)

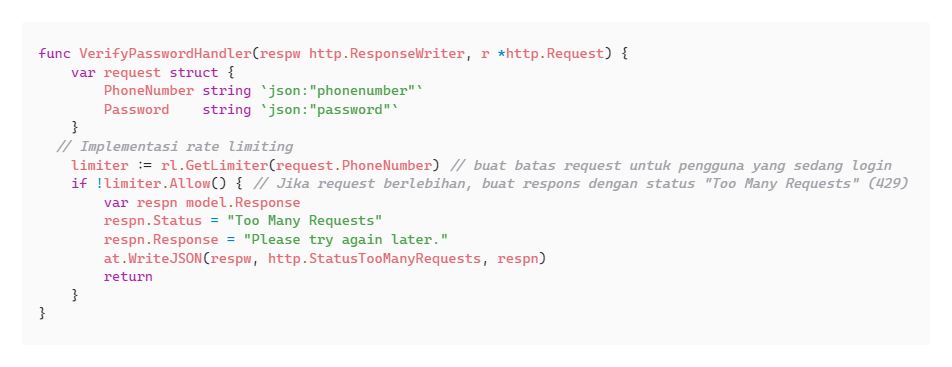
Penanganan permintaan otentikasi dengan baik adalah kunci untuk menjaga keamanan aplikasi. Sistem yang tidak memiliki kontrol dan pemantauan yang memadai terhadap permintaan dapat menjadi sasaran mudah bagi serangan. Di bagian ini, kita akan melihat bagaimana program yang lemah dan kuat menangani permintaan otentikasi.

Program yang lemah terhadap aspek ini adalah program yang tidak memiliki mekanisme untuk membatasi jumlah permintaan yang dapat dilakukan dalam jangka waktu tertentu, membuatnya rentan terhadap serangan brute force.



**Gambar 14 Ilustrasi program yang tidak memiliki rate limiter**

Sementara itu, program yang digunakan oleh penelitian ini, sudah mengimplementasikan mekanisme rate limiting untuk membatasi jumlah permintaan otentikasi dalam jangka waktu tertentu.



**Gambar 15 Program yang menerapkan mekanisme rate limiter**

1. **Keterbatasan** **dan** **Tantangan**

Salah satu keterbatasan adalah ketergantungan pada platform WhatsApp untuk pengiriman STP. Jika pengguna tidak memiliki akses ke WhatsApp, proses autentikasi mungkin menjadi terhambat.

Implementasi sistem yang menggabungkan tiga metode autentikasi memerlukan upaya teknis yang lebih besar dibandingkan dengan sistem autentikasi tunggal. Namun, manfaat keamanan yang diperoleh dianggap sepadan dengan upaya tersebut.

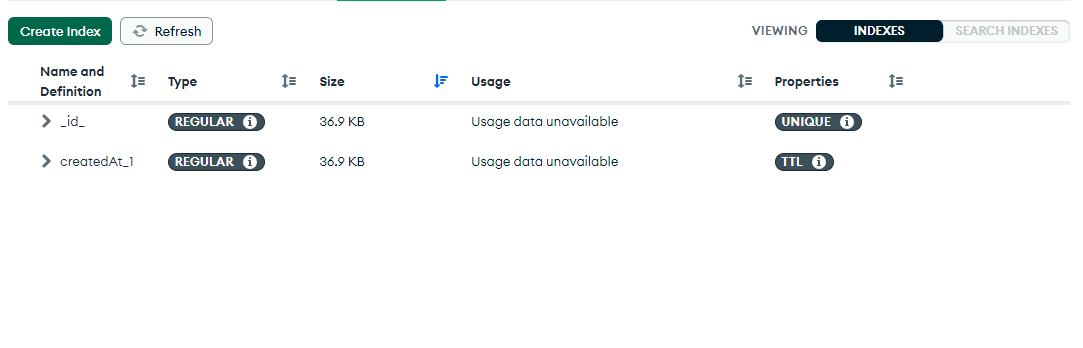
Pengujian menggunakan metode penetration test dengan teknik brute force dan dictionary attack, yang berfokus pada satu metode autentikasi yaitu short temporary password (STP). Meskipun penetration test ini dirancang untuk mengidentifikasi potensi kelemahan dalam sistem autentikasi STP, fokus yang terbatas pada satu metode autentikasi saja berarti hasil pengujian mungkin tidak mencerminkan keamanan sistem secara keseluruhan. Selain itu, teknik brute force dan dictionary attack memiliki keterbatasan karena hanya mampu menguji kombinasi karakter dan daftar kata yang telah ditentukan, sehingga tidak dapat menjamin bahwa semua kemungkinan celah keamanan telah ditemukan. Oleh karena itu, meskipun pengujian ini memberikan wawasan berharga tentang ketahanan metode STP terhadap serangan tertentu, tetap diperlukan pengujian tambahan dengan teknik dan skenario berbeda untuk memastikan keamanan yang komprehensif.

## 4. KESIMPULAN

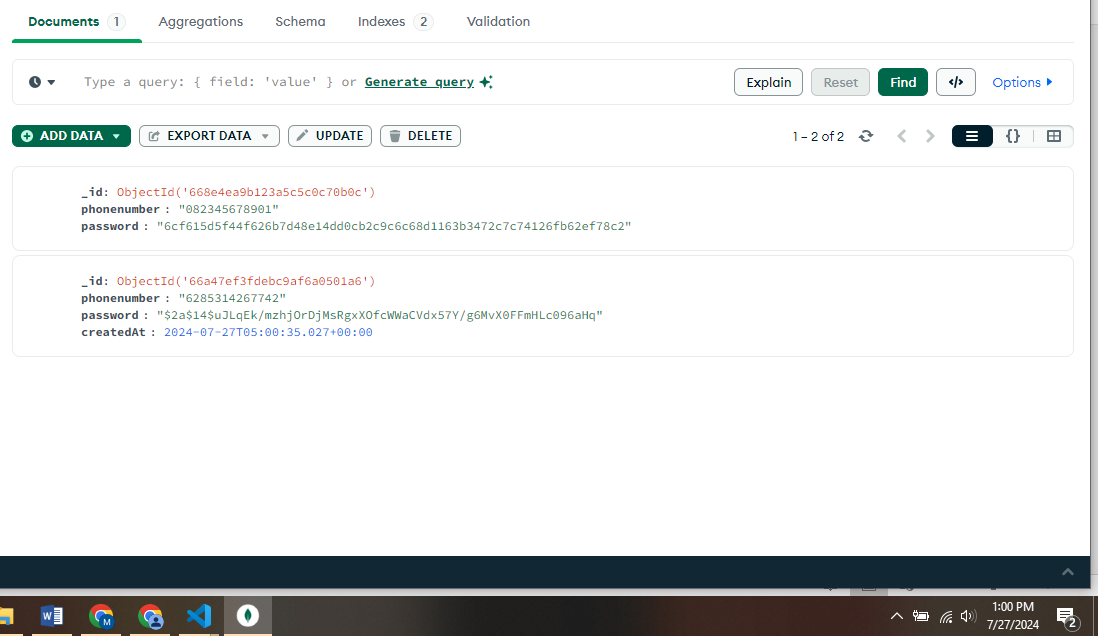
Penggabungan tiga metode autentikasi, yaitu QR code, OAuth2 (Google), dan short temporary password (STP), secara signifikan meningkatkan keamanan autentikasi pada web statis. Implementasi metode STP yang menggunakan bcrypt untuk hashing kata sandi dan MongoDB sebagai basis data dengan pengindeksan pada field createdAt, memastikan bahwa kata sandi sementara otomatis hilang setelah 4 menit, memberikan lapisan perlindungan tambahan. Validasi CAPTCHA, sistem rate limiting, dan penggunaan sistem deteksi intrusi juga efektif dalam mencegah serangan otomatis dan aktivitas mencurigakan. Pengujian brute force dan dictionary attack selama 5 menit masing-masing menunjukkan bahwa teknik-teknik ini tidak berhasil menembus sistem, menegaskan efektivitas mekanisme keamanan yang diimplementasikan. Keseluruhan hasil penelitian ini menunjukkan bahwa pendekatan autentikasi berlapis ini dapat diadopsi oleh pengembang web statis untuk meningkatkan perlindungan terhadap data dan pengguna, serta memberikan solusi inovatif untuk mengatasi kelemahan yang ada pada pendekatan autentikasi konvensional.

Untuk memastikan keamanan yang optimal dalam sistem otentikasi, sangat penting untuk:

1. Menggunakan hashing untuk menyimpan kata sandi.
2. Melakukan pengindeksan pada setiap data, yang merupakan salah satu fitur MongoDB, agar data autentikasi yang tersimpan dapat dihapus secara berkala. Hal ini juga memastikan keamanan dari upaya pembobolan akun, meskipun akun dicoba ditebak password-nya berkali-kali menggunakan Teknik brute force. Berikut gambar yang menunjukkan konfigurasi dari collection yang digunakan autentikasi, dimana field createdAt yang diberikan indeks selama 5 menit.

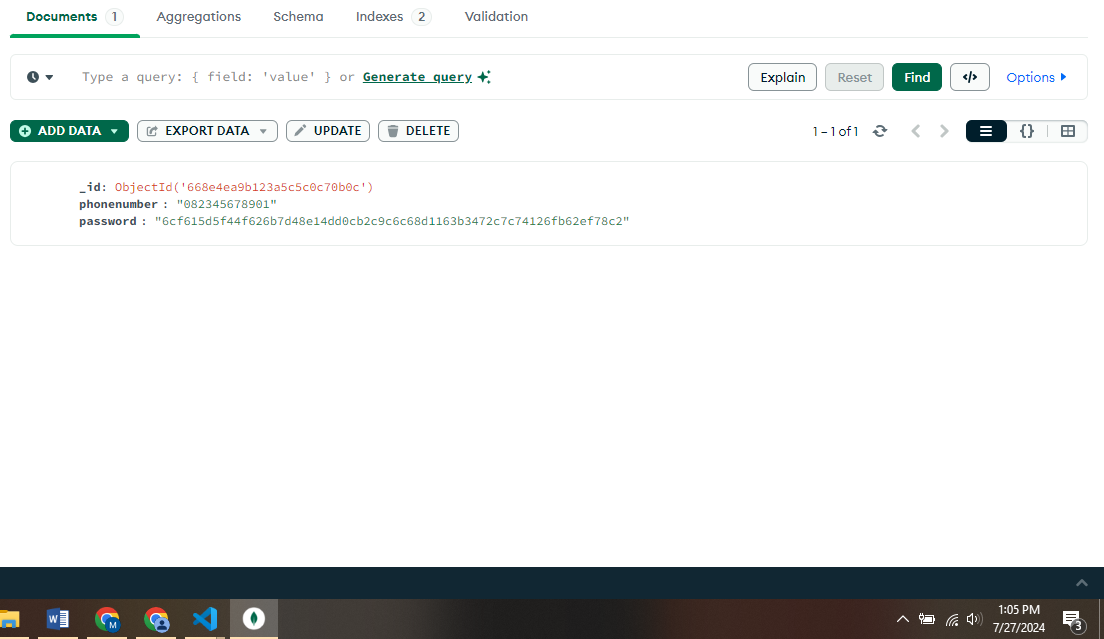


**Gambar 16 Konfigurasi indeks MongoDb**



**Gambar 17 Database ketika ada proses autentikasi**

Ini memungkinkan, setiap adanya proses autentikasi pengguna, data hanya akan disimpan dalam waktu 5 menit, setelah tenggang waktu tersebut, maka data akan otomatis hilang, sehingga password tidak akan lagi valid, dan mencegah adanya upaya pembobolan akun.



**Gambar 18 Database setelah 5 menit dari proses autentikasi**

1. Mengintegrasikan CAPTCHA untuk memverifikasi bahwa permintaan berasal dari manusia.
2. Menerapkan batasan pada jumlah percobaan login untuk mencegah serangan brute force
3. Melakukan validasi dan sanitasi input dengan benar.

Dengan mengikuti praktik terbaik ini, aplikasi akan lebih terlindungi dari berbagai ancaman keamanan, termasuk serangan brute force dan serangan berbasis web lainnya.

## UCAPAN TERIMA KASIH

Segala puji bagi Allah Yang Maha Esa, atas limpahan berkat dan rahmat-Nya, penulis berhasil menyelesaikan laporan ini dengan baik. Penulis juga ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Orang tua dan rekan-rekan yang senantiasa memberikan doa dan dukungan, sehingga penulis dapat melalui setiap tantangan dengan penuh semangat.
2. Bapak Rolly Maulana Awangga, S.T., MT., CAIP, SFPC selaku pembimbing internal, yang telah memberikan bimbingan, arahan, dan motivasi sepanjang proses penulisan laporan ini.
3. Bapak M. Yusril Helmi Setyawan, S. Kom., M. Kom., SFPC selaku pembimbing internal, yang telah menyediakan waktu, pengetahuan, dan keahlian dalam membantu penulis menyelesaikan penelitian ini.
4. Bapak Roni Andarsyah, S.T., M. Kom., SFPC selaku Kepala Program Studi DIV Teknik Informatika Universitas Logistik Bisnis Internasional, atas dukungan dan kebijaksanaannya dalam memfasilitasi proses penelitian ini.
5. Bapak M. Yusril Helmi Setyawan, S. Kom., M. Kom., SFPC selaku Koordinator tugas akhir tahun 2024/2025, yang telah memberikan panduan dan koordinasi yang sangat membantu dalam penyelesaian tugas akhir ini.

Ucapan terima kasih ini juga ditujukan kepada semua pihak yang tidak dapat disebutkan satu per satu, tetapi telah memberikan kontribusi yang berarti dalam penyelesaian laporan ini. Semoga segala kebaikan dan bantuan yang diberikan mendapatkan balasan yang setimpal dari Allah SWT.

## DAFTAR rujukan

[1] J. Shahid, M. K. Hameed, I. T. Javed, K. N. Qureshi, M. Ali, and N. Crespi, “A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions,” *Applied Sciences*, vol. 12, no. 8, 2022, doi: 10.3390/app12084077.

[2] “Cybercrime To Cost The World $10.5 Trillion Annually By 2025.” Accessed: Jul. 22, 2024. [Online]. Available: https://cybersecurityventures.com/cybercrime-damage-costs-10-trillion-by-2025/

[3] B. Narwal and A. K. Mohapatra, “A survey on security and authentication in wireless body area networks,” *Journal of Systems Architecture*, vol. 113, p. 101883, 2021, doi: https://doi.org/10.1016/j.sysarc.2020.101883.

[4] D. V Kornienko, S. V Mishina, and M. O. Melnikov, “The Single Page Application architecture when developing secure Web services,” in *Journal of Physics: Conference Series*, 2021, p. 12065.

[5] M. Eminagaoglu, E. Cini, G. Sert, and D. Zor, “A Two-Factor Authentication System with QR Codes for Web and Mobile Applications,” in *2014 Fifth International Conference on Emerging Security Technologies*, 2014, pp. 105–112. doi: 10.1109/EST.2014.19.

[6] A. Arvind, P. Mahajan, and R. Chalke, “TOTP Based Authentication Using QR Code For Gateway Entry System,” *International Journal of Engineering and Computer Science*, vol. 9, pp. 25023–25028, Jul. 2020, doi: 10.18535/ijecs/v9i05.4481.

[7] C. Sudar, S. K. Arjun, and L. R. Deepthi, “Time-based one-time password for Wi-Fi authentication and security,” in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1212–1216. doi: 10.1109/ICACCI.2017.8126007.

[8] R. Yang, W. Lau, and S. Shi, “Breaking and Fixing Mobile App Authentication with OAuth2.0-based Protocols,” Jul. 2017, pp. 313–335. doi: 10.1007/978-3-319-61204-1\_16.

[9] D. Nolan and D. Temple Lang, “Authentication for Web Services via OAuth,” 2014, pp. 441–461. doi: 10.1007/978-1-4614-7900-0\_13.

[10] D.-S. Oh, B.-H. Kim, and J.-K. Lee, “A Study on Authentication System Using QR Code for Mobile Cloud Computing Environment,” vol. 184, Jul. 2011, doi: 10.1007/978-3-642-22333-4\_65.

[11] M. Alizadeh, S. Abolfazli, M. Zamani, S. Baharun, and K. Sakurai, “Authentication in mobile cloud computing: A survey,” *Journal of Network and Computer Applications*, vol. 61, pp. 59–80, 2016, doi: https://doi.org/10.1016/j.jnca.2015.10.005.

[12] D. Mahansaria and U. K. Roy, “Secure Authentication Using One Time Contextual QR Code,” in *International Symposium on Security in Computing and Communications*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:219008525

[13] C. E. (Catherine) Lee, H. H. Chern, and D. A. Azmir, “WhatsApp Use in a Higher Education Learning Environment: Perspective of Students of a Malaysian Private University on Academic Performance and Team Effectiveness,” *Educ Sci (Basel)*, vol. 13, no. 3, 2023, doi: 10.3390/educsci13030244.

[14] M. Bala Krishna and A. Dugar, “Product Authentication Using QR Codes: A Mobile Application to Combat Counterfeiting,” *Wirel Pers Commun*, vol. 90, no. 1, pp. 381–398, 2016, doi: 10.1007/s11277-016-3374-x.

[15] A. Mühle, A. Grüner, T. Gayvoronskaya, and C. Meinel, “A survey on essential components of a self-sovereign identity,” *Comput Sci Rev*, vol. 30, pp. 80–86, 2018, doi: https://doi.org/10.1016/j.cosrev.2018.10.002.

[16] V. Grover and others, “An Efficient Brute Force Attack Handling Techniques for Server Virtualization,” in *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*, 2020.

[17] T. Gautam and U. Singh, “AN APPROACH FOR DETECTING PASSWORD PATTERN IN DICTIONARY ATTACK,” 2022.