MULTI FACTOR AUTHENTICATION PADA E-COMMERCE PENJUALAN BATIK DENGAN HASHING ALGORITMA BLOWFISH, ONE-TIME PASSWORD, DAN AES 256

T.Firmansyah[1], R.Renaldy[2] dan A.Z.Alfurqon[3]

[1,2,3]*Jurusan Informatika, Fakultas Teknik dan Informatika, Universitas PGRI Semarang*

Gedung Pusat Lantai 3, Kampus 1 Jl. Sidodadi Timur 24, Semarang

E-mail : tedysyh07@gmail.com[1], ramadhanrenaldy@upgris.ac.id[2], zakyysme@gmail.com[3]

Abstrak

Keamanan adalah aspek penting dalam sistem e-commerce, terutama dalam mengatasi ancaman siber seperti pencurian data, peretasan, dan akses tidak sah. Penelitian ini bertujuan untuk mengimplementasikan metode keamanan berbasis Multi-Factor Authentication (MFA) pada sistem e-commerce penjualan batik, dengan mengintegrasikan hashing Blowfish, enkripsi Advanced Encryption Standard (AES)-256, dan One-Time Password (OTP). Pendekatan ini dirancang untuk meningkatkan perlindungan data pengguna serta memastikan keamanan transaksi. Proses sistem dimulai dengan registrasi, di mana password di-hash menggunakan algoritma Blowfish. Pada tahap login, pengguna harus memasukkan OTP yang dikirim melalui saluran aman, seperti email atau Short Message Service (SMS). Setelah validasi, pengguna membuat Personal Identification Number (PIN) yang dienkripsi menggunakan algoritma AES-256 sebelum disimpan dalam database. Hasil implementasi menunjukkan kombinasi hashing Blowfish, enkripsi AES-256, dan OTP memberikan tingkat keamanan signifikan terhadap data pengguna dan transaksi. Sistem berhasil melindungi dari ancaman seperti serangan brute force dan akses tidak sah. Namun, ditemukan kendala teknis, seperti kecepatan hashing Blowfish dan kebutuhan sumber daya tinggi untuk AES-256. Dengan penerapan yang tepat, metode ini dapat mendukung pengembangan sistem e-commerce yang lebih aman, memberikan perlindungan menyeluruh bagi data pengguna, dan meningkatkan kepercayaan serta kenyamanan dalam bertransaksi secara digital.

Kata Kunci**:** *Multi-Factor Authentication, Blowfish,* AES-256*, One-Time Password, E-Commerce.*

I. PENDAHULUAN

Berbagai inovasi teknologi muncul dengan cepat dan menimbulkan banyak kejutan di kalangan masyarakat pada era modern. Sebagian besar teknologi tersebut memberikan dampak yang signifikan pada berbagai aspek kehidupan manusia, termasuk di sektor *e-commerce*. *E-commerce* merupakan salah satu kegiatan perdagangan yang dilakukan secara online dengan memanfaatkan internet dalam bisnisnya [1]. *E-commerce* juga diartikan sebagai bentuk pasar digital yang tidak memiliki bentuk fisik, di mana semua transaksi dilakukan secara daring.

Kemajuan pesat teknologi informasi dan komunikasi, terutama melalui e-commerce, telah secara signifikan mengubah lanskap dunia bisnis. *e-commerce* memberikan

E-ISSN: 2985-3532

peluang bagi Usaha Mikro, Kecil, dan Menengah (UMKM) untuk memperluas pasar, meningkatkan efisiensi operasional, dan mendapatkan manfaat ekonomi yang lebih besar [2]. Akhir-akhir ini, perhatian terhadap peran e-commerce dalam mendorong pertumbuhan ekonomi UMKM semakin meningkat.

Namun, seiring dengan perkembangan pesat e-commerce, isu keamanan sistem menjadi salah satu aspek yang sangat penting untuk diperhatikan. Tantangan keamanan yang dihadapi dalam transaksi *e-commerce* mencakup ancaman yang beragam, seperti pencurian identitas, serangan malware, serangan phishing, dan pelanggaran data [3]. Ancaman akses tidak sah yang dapat membahayakan data dan informasi pengguna, serta potensi penyalahgunaan sistem.

Serangan siber, seperti peretasan dan pencurian data, dapat menimbulkan kerugian yang signifikan bagi penyedia layanan e-commerce maupun konsumennya. Oleh karena itu, penerapan metode keamanan yang andal menjadi kebutuhan yang mendesak dalam sistem *e-commerce*.

Dalam upaya mengamankan sistem *e-commerce*, terdapat berbagai metode yang dapat digunakan untuk melindungi data pengguna dan transaksi, salah satunya adalah *Multi-Factor Authentication* (MFA). MFA merupakan mekanisme autentikasi yang memerlukan lebih dari satu faktor verifikasi untuk memastikan identitas pengguna.

Selain MFA, penerapan kriptografi juga memainkan peran penting dalam pengamanan data pada sistem *e-commerce*. Kriptografi adalah ilmu menjaga kerahasiaan pesan dengan cara menjadikannya dalam bentuk yang tidak dapat dipahami lagi [7]. Salah satu teknik kriptografi yang umum digunakan adalah *hashing* dengan algoritma *blowfish*.

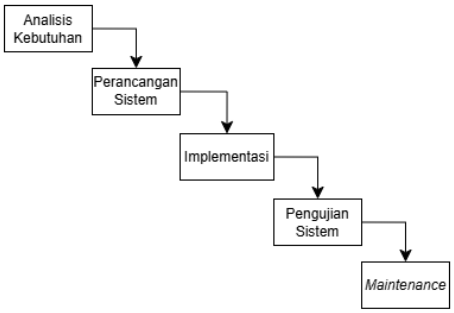
Algoritma Blowfish adalah salah satu algoritma kriptografi simetris yang dirancang oleh Bruce Schneier pada tahun 1993 untuk menyediakan alternatif yang cepat dan aman dari algoritma enkripsi lainnya, seperti *Data Encryption Standard* (DES). Algoritma ini menggunakan panjang kunci variabel antara 32 bit hingga 448 bit, yang memberikan fleksibilitas dalam tingkat keamanan.

Untuk enkripsi data, algoritma *Advanced Encryption Standard* (AES) 256 sering digunakan karena keamanannya yang tinggi dengan kunci 256 bit. AES 256 memastikan data terenkripsi kuat dan hanya dapat diakses oleh pihak dengan kunci dekripsi. Kombinasi *Multi Factor Authentication* (MFA), hashing dengan algoritma *blowfish*, OTP, dan algoritma AES 256 membantu melindungi e-commerce dari ancaman siber seperti pencurian data atau akses tidak sah.

II. METODOLOGI PENELITIAN

28. Metodologi Penelitian

E-ISSN: 2985-3532



Gambar 1. Sistem *Waterfall*.

Metode *waterfall* digunakan dalam penelitian "*Multi Factor Authentication* Pada *E-Commerce* Penjualan Batik dengan Hashing Algoritma *Blowfish*, *One-Time Password*, dan AES 256". Model Waterfall adalah Model Air Terjun kadang dinamakan siklus hidup klasik dimana dilakukan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak. Metode ini dilakukan dengan pendekatan yang sistematis [5]. Langkah-langkah dalam penerapan metode waterfall ini dilakukan melalui lima tahapan yaitu *requirement analyst*, *design*, *implementation*, *testing*, serta pemeliharaan [6].

1.1. Analisa Kebutuhan

Tahap analisis kebutuhan merupakan langkah awal dalam proses pengembangan sistem. Pada tahap ini, dilakukan identifikasi dan analisis terhadap kebutuhan yang diperlukan untuk mendukung penelitian *"Multi Factor Authentication pada E-Commerce Penjualan Batik dengan Hashing Algoritma Blowfish, One-Time Password, dan AES 256"*. Kebutuhan yang dianalisis mencakup kebutuhan fungsional dan non-fungsional sistem.

1.2. Perancangan Sistem

Tahap perancangan sistem bertujuan untuk membuat rancangan yang menjadi acuan dalam pengembangan sistem sesuai dengan hasil analisis kebutuhan. Pada tahap ini, dilakukan desain arsitektur sistem, alur proses autentikasi multi faktor, serta model data yang mendukung integrasi algoritma *bowfish*, OTP, dan AES 256. Selain itu, dibuat pula rancangan antarmuka pengguna yang intuitif serta diagram alur kerja sistem untuk memberikan gambaran yang jelas mengenai operasional sistem. Semua rancangan ini diharapkan dapat meminimalkan risiko kegagalan pada tahap implementasi.

1.3. Implmentasi

Tahap implementasi merupakan tahap dimana pada tahap tersebut akan melakukan aksi untuk menerapkan apa yang sudah direncanakan [7]. Pada tahap ini, algoritma *hashing Blowfish* diintegrasikan untuk melindungi data autentikasi pengguna, mekanisme OTP diimplementasikan untuk memberikan lapisan keamanan tambahan, dan algoritma AES 256 digunakan untuk mengenkripsi data sensitif. Proses implementasi melibatkan pengembangan backend dengan teknologi yang relevan serta integrasi dengan modul *frontend* untuk memastikan sistem dapat berjalan sesuai rancangan.

1.4. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa aplikasi yang dikembangkan telah memenuhi kebutuhan fungsional dan non-fungsional yang ditentukan. Tahapan ini mencakup pengujian fungsional menggunakan metode *blackbox* testing untuk memvalidasi setiap fitur utama, seperti pembuatan OTP, proses autentikasi, dan

E-ISSN: 2985-3532

enkripsi data. Selain itu, dilakukan pengujian keamanan untuk mengevaluasi ketahanan sistem terhadap ancaman, seperti serangan *brute force* atau pencurian data. Hasil pengujian ini menjadi dasar untuk mengevaluasi apakah sistem layak digunakan atau memerlukan perbaikan lebih lanjut.

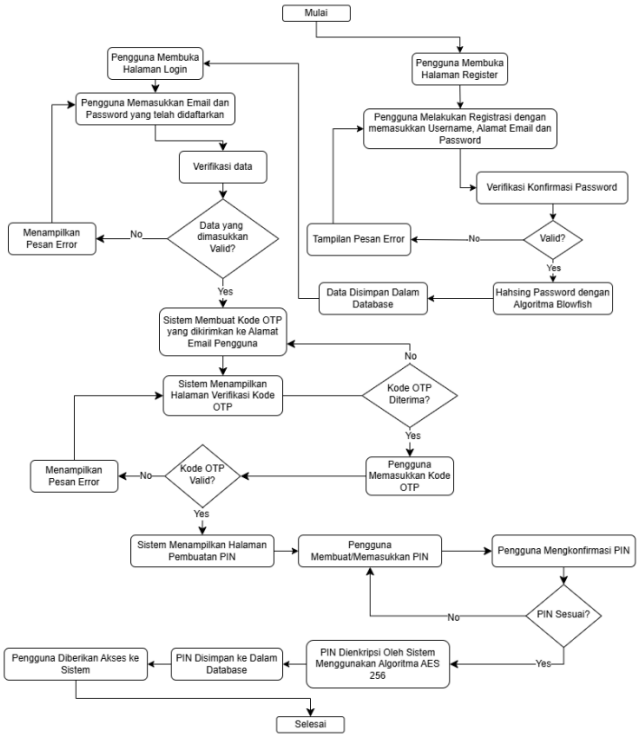
1.5. *Maintenance*

Tahap pemeliharaan sistem bertujuan untuk memastikan sistem dapat berjalan dengan baik dalam jangka panjang. Aktivitas pemeliharaan meliputi pembaruan sistem guna meningkatkan keamanan, perbaikan terhadap bug yang ditemukan selama penggunaan, serta penyesuaian terhadap kebutuhan baru yang mungkin muncul di masa depan. Pemeliharaan ini dilakukan secara berkala untuk menjaga keandalan dan performa sistem, sekaligus memastikan pengguna mendapatkan pengalaman terbaik saat menggunakan aplikasi.

III. HASIL DAN PEMBAHASAN

1. ANALISA KEBUTUHAN

Dari analisa yang telah dilakukan, dibutuhkan kebutuhan fungsional dan non-fungsional pada penilitian ini. Kebutuhan fungsional meliputi mekanisme autentikasi multi faktor, implementasi algoritma *hashing Blowfish*, pengiriman *One-Time Password* (OTP), serta enkripsi data menggunakan AES 256. Sedangkan kebutuhan non-fungsional mencakup aspek keamanan, keandalan, dan performa sistem untuk menjamin pengalaman pengguna yang optimal.



Gambar 2. *Flowchart Diagram.*

Pada gambar 2. menjelaskan cara kerja sistem "*Multi Factor Authentication* Pada *E-Commerce* Penjualan Batik dengan Hashing Algoritma *Blowfish*, *One-Time Password*, dan AES 256" dengan langkah-langkah:

1. Pengguna memasukkan username dan password.
2. Password di-hash menggunakan algoritma *Blowfish*.
3. Sistem memverifikasi username dan hash password di database.
4. Jika autentikasi berhasil, sistem akan membuat kode OTP (*One-Time Password*).
5. OTP terenkripsi dikirim ke perangkat pengguna (misalnya melalui SMS atau email).
6. Pengguna memasukkan OTP yang diterima.
7. Jika OTP valid, maka sistem akan menampilkan halaman untuk mengatur pin yang nantinya berguna untuk transaksi.
8. Jika pin yang dimasukkan sesuai, oleh sistem, pin akan dienkripsi menggunakan *Advanced Encryption Standard* (AES) 256 dan akan disimpan dalam database.

2.1. Algoritma *Blowfish*

Blowfish mengenkripsi data dalam blok 64 bit menggunakan struktur Feistel dengan 16 putaran, melibatkan substitusi, XOR, dan pergeseran bit. Algoritma ini efisien dalam pengolahan pada perangkat keras dan perangkat lunak, sehingga cocok untuk aplikasi berkecepatan tinggi. Meski telah digantikan algoritma modern seperti AES, *Blowfish* masih digunakan dalam berbagai sistem keamanan data dan komunikasi. Algoritma ini mengubah data menjadi format yang sulit dipahami melalui langkah-langkah:

a. Data dibagi menjadi blok-blok berukuran 64-bit.

b. Setiap blok diproses melalui struktur Feistel dengan 16 putaran.

c. Pada setiap putaran, dilakukan operasi substitusi, XOR, dan pergeseran bit

menggunakan subkunci yang dihasilkan dari kunci utama.

d. Hasil akhir adalah data terenkripsi yang tidak dapat dibaca tanpa kunci dekripsi.

Perhitungan :

Input: Misalkan kita ingin mengenkripsi plaintext 64-bit yang terbagi menjadi dua bagian, L dan R (32-bit masing-masing).

Kunci: Kunci 256-bit digunakan untuk mengisi P-array dan S-boxes.

Putaran 1:

E-ISSN: 2985-3532

- Fungsi Feistel dihitung dengan menggunakan R (bagian kanan) dan sub-kunci P1.
- Hasil dari fungsi Feistel di-XOR-kan dengan L (bagian kiri).
- L dan R dipertukarkan.

Putaran 2 sampai 16: Proses yang sama dilakukan dengan menggunakan sub-kunci P2 hingga P16.

Output: Setelah 16 putaran, gabungkan L dan R untuk menghasilkan ciphertext 64-bit.

2.2. *One-Time Password* (OTP)

One-Time Password (OTP) merupakan mekanisme autentikasi dinamis yang menghasilkan kata sandi sekali pakai untuk memverifikasi identitas pengguna dalam sistem keamanan berbasis Multi-Factor Authentication (MFA). Dalam penelitian ini, OTP digunakan untuk menambahkan lapisan keamanan tambahan pada sistem e-commerce penjualan batik, sehingga mengurangi risiko akses tidak sah atau serangan siber seperti *phishing* dan *replay attack*.

OTP bersifat sementara, hanya berlaku untuk satu sesi autentikasi atau dalam jangka waktu tertentu, sehingga sulit untuk ditebak atau digunakan kembali oleh pihak yang tidak berwenang. Algoritma OTP bekerja dengan langkah-langkah berikut:

a. Sistem menghasilkan kode OTP unik berdasarkan algoritma tertentu, seperti *Time-Based One-Time Password* (TOTP) atau *HMAC-Based One-Time Password* (HOTP).

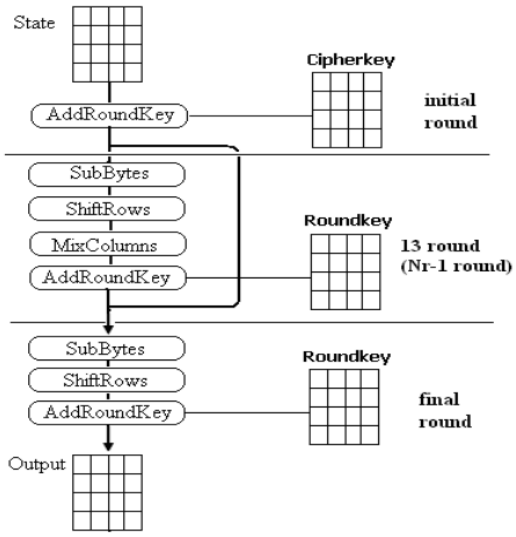
b. Kode OTP dikirim ke pengguna melalui saluran aman, seperti email atau pesan singkat (SMS).

c. Pengguna memasukkan kode OTP yang diterima ke dalam sistem sebagai bagian dari proses autentikasi.

d. Sistem memvalidasi OTP yang dimasukkan dengan membandingkannya dengan kode yang dihasilkan pada saat itu. Jika valid, akses diberikan, namun jika tidak, autentikasi ditolak.

2.3. AES-256 (*Advanced Encryption Standard* 256-bit)

E-ISSN: 2985-3532



Gambar 3. Proses Enkripsi *Advanced Encryption Standard* (AES).

AES adalah algoritma enkripsi aman untuk melindungi data atau informasi sensitif menggunakan berbagai teknik enkripsi dan dekripsi panjang kunci seperti 128-bit, 192 bit, dan 256 bit. Pada tahun 2001, Pada proses enkripsi AES menggunakan 4 transformasi dasar dengan urutan transformasi subbytes, shiftrows, mixcolumns, dan addroundkey. Sedangkan Pada proses dekripsi menggunakan invers semua transformasi dasar pada algoritma AES kecuali addroundkey dengan urutan transformasi invshiftrows, invsubbytes, addroundkey,dan invmixcolumns [7].

Algoritma enkripsi simetris yang menggunakan kunci 256-bit. Digunakan untuk mengenkripsi dan mendekripsi PIN. Dengan langkah-langkah:

a. Generate kunci enkripsi 256-bit.

b. Teks OTP dibagi menjadi blok-blok 128-bit.

c. Setiap blok dienkripsi melalui beberapa putaran transformasi.

d. Hasil enkripsi digabungkan menjadi ciphertext.

e. Untuk dekripsi, proses dibalik menggunakan kunci yang sama.

Perhitungan :

1. Kunci: Kunci 256-bit (misalnya, 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f).
2. Plaintext: Teks yang ingin dienkripsi (hello1234567890).
3. Key Expansion: Kunci 256-bit diproses menjadi 60 sub-kunci 32-bit.

E-ISSN: 2985-3532

Putaran:

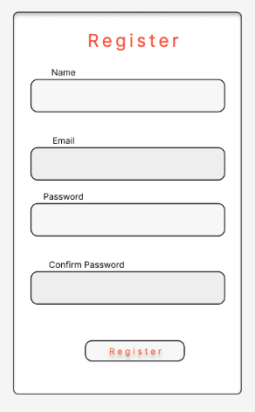
- Putaran pertama hingga ke-13: Melibatkan *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey*.

- Putaran ke-14: Hanya *SubBytes*, *ShiftRows*, dan *AddRoundKey* tanpa *MixColumns.*

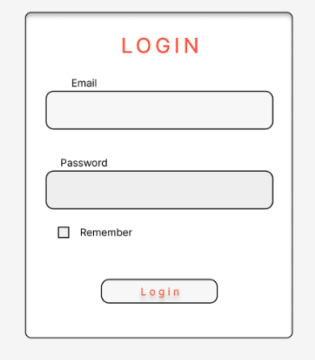
Ciphertext: Hasil dari enkripsi adalah ciphertext yang tidak dapat langsung dikenali tanpa kunci.

2.3. Gambaran Sistem

Gambaran sistem diperlukan sebelum perancangan suatu sistem. Fungsi utama dari gambaran sistem adalah untuk menyajikan kerangka konseptual yang jelas, sehingga memudahkan proses identifikasi kebutuhan, batasan, serta tujuan sistem. Gambaran sistem dapat berupa *mockup/user interface* sistem. *User interface* dari perancangan sistem "*Multi Factor Authentication* Pada *E-Commerce* Penjualan Batik dengan Hashing Algoritma *Blowfish*, *One-Time Password*, dan AES 256" terdapat pada gambar 4-7.



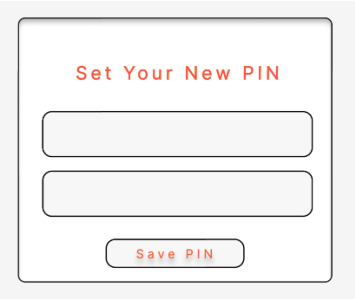
Gambar 4. User Interface *Register*.



Gambar 5. *User Interface Login.*

E-ISSN: 2985-3532

Gambar 6. *User Interface* Verifikasi OTP.



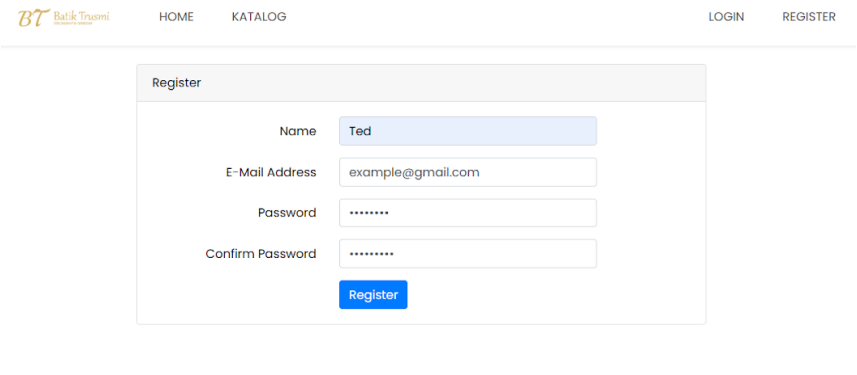
Gambar 7. *User Interface* Pembuatan PIN.

3. IMPLEMENTASI

3.1 Implementasi Sistem

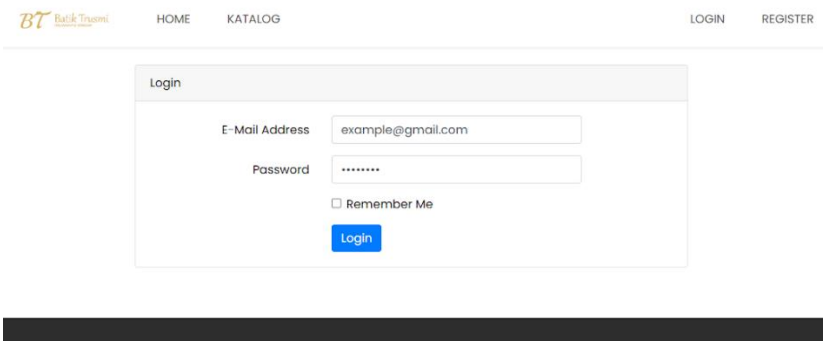
Setelah melewati tahap perancangan sistem, peneliti berhasil mengimplementasi metode yang telah ditentukan ke dalam sistem. Pada gambar 8-14 adalah dokumentasi dari hasil implementasi metode ke dalam sistem *e-commerce* penjualan batik.

E-ISSN: 2985-3532



Gambar 8. Halaman *Register*.

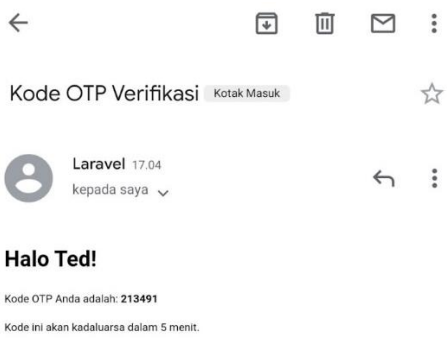
Pada halaman *Register* terdapat form pengisian nama, alamat email, password dan konfirmasi password. Setelah pengguna mengisi semua form untuk mendaftar dengan valid dan pengguna meng-klik tombol register maka sistem akan menyimpan data dan melakukan hashing password menggunakan algoritma *Blowfish* ke dalam database.



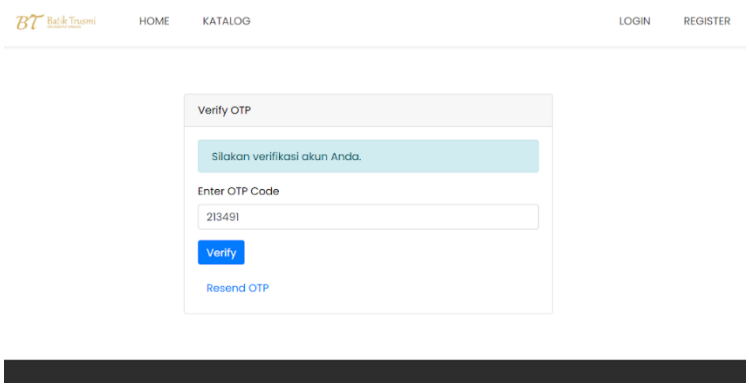
Gambar 9. Halaman *Login*.

Setelah melakukan registrasi, pengguna diarahkan ke halaman login, pengguna dapat memasukkan alamat email dan password, dan melakukan login. Pada tahap pengguna melakukan login, sistem akan membuat *One-Time Password* (OTP) yang akan dikirimkan melalui pesan email.

E-ISSN: 2985-3532



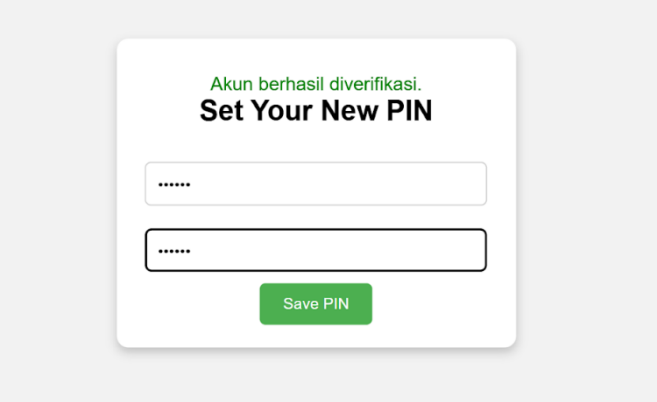
Gambar 10. Pesan pada Email diterimanya *One-Time Password* (OTP).



Gambar 11. Halaman Autentikasi dengan OTP.

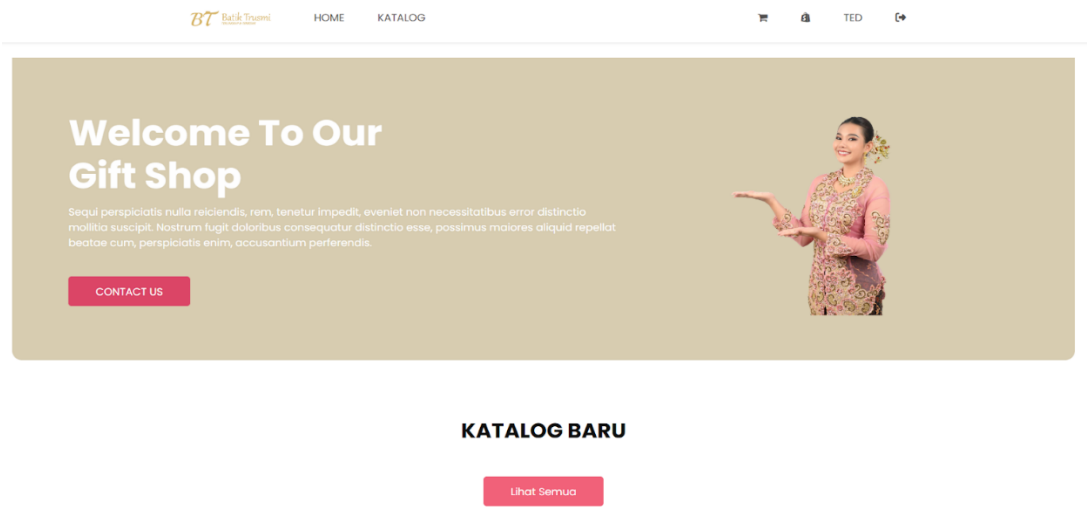
Setelah *One-Time Password* (OTP) dibuat oleh sistem, sistem akan menampilkan halaman untuk memasukkan *One-Time Password* yang diterima oleh pengguna. Setelah pengguna mendapatkan pesan kode OTP, pengguna dapat memasukkan kode OTP tersebut ke dalam form halaman verifikasi OTP. Jika valid, maka sistem akan memberikan akses masuk sistem ke tahap selanjutnya.

E-ISSN: 2985-3532



Gambar 12. Halaman Pembuatan PIN.

Pada Halaman pembuatan PIN yang berguna untuk keamanan dalam melakukan transaksi, pengguna dapat menggunakan angka pin yang berjumlah enam karakter. Jika PIN yang dimasukkan valid, sistem akan melakukan enkripsi menggunakan *Advanced Encryption Standard* (AES) 256 dan disimpan ke dalam database.

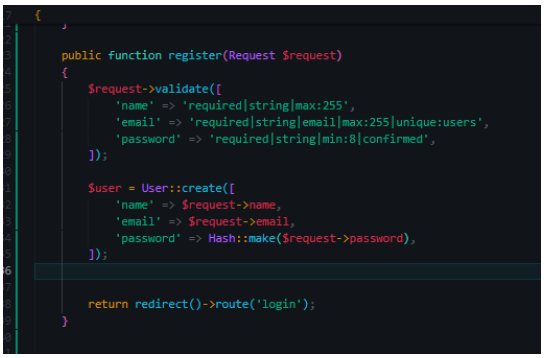


Gambar 13. Halaman Pembuatan PIN.

Jika semua tahapan telah selesai, pengguna dapat mengakses sistem *e-commerce* penjualan batik dan melakukan transaksi di dalamnya.

2.2. Implementasi Algoritma dan *One-Time Password*

E-ISSN: 2985-3532



```
{
    public function register(Request $request)
    {
        $request->validate([
            'name' => 'required|string|max:255',
            'email' => 'required|string|email|max:255|unique:users',
            'password' => 'required|string|min:8|confirmed',
        ]);

        $user = User::create([
            'name' => $request->name,
            'email' => $request->email,
            'password' => Hash::make($request->password),
        ]);


        return redirect()->route('login');
    }
```

Gambar 14. Implementasi *Hashing Blowfish*.

Pada Implementasi sistem, Metode *hashing blowfish* digunakan untuk membuat password agar menjadi sebuah kalimat atau huruf acak agar tidak mudah dibobol. Proses *hashing blowfish* bekerja dengan mengenkripsi pesan (atau data) menggunakan kunci yang telah ditentukan, dan hasil enkripsi *(ciphertext)* dapat digunakan sebagai nilai *hash*.

Proses enkripsi password akun dengan algoritma *blowfish* mengikuti struktur *Feistel*, di mana data dibagi menjadi dua bagian: kiri (L) dan kanan (R). Setiap bagian ini kemudian diproses dalam beberapa putaran (16 putaran untuk *blowfish*). Pada setiap putaran, bagian kanan (R) diubah menggunakan fungsi yang melibatkan *P-array* dan *S-boxes*, yang berisi angka-angka yang dihasilkan dari kunci yang diberikan.

Setelah itu, hasilnya akan di-XOR-kan dengan bagian kiri (L) dan kedua bagian tersebut dipertukarkan. Proses ini berlanjut hingga semua putaran selesai. Pada akhir proses, bagian kiri dan kanan yang telah diproses digabungkan untuk menghasilkan *ciphertext*, yang merupakan data terenkripsi yang hanya dapat didekripsi kembali dengan kunci yang tepat.

E-ISSN: 2985-3532

```php
class PinController extends Controller
{
    public function store(Request $request)
    {
        $request->validate([
            'new_pin' => 'required|min:4|max:6|same:confirm_pin',
            'confirm_pin' => 'required',
        ]);

        $encryptedPin = $this->encryptPin($request->new_pin);

        $user = Auth::user(); // Pastikan pengguna sudah login
        $user->pin = $encryptedPin;
        $user->save();

        // Simpan PIN ke database atau lakukan tindakan lain
        // auth()->user()->update(['pin' => bcrypt($request->new_pin)]);

        return redirect()->route('home')->with('message','Pin Berhasil Disimpan!');
    }

    protected function encryptPin($pin)
    {
        return Crypt::encryptString($pin);
    }

    protected function decryptPin($encryptedPin)
{
    return Crypt::decryptString($encryptedPin);
```

Gambar 15. Implementasi *Advanced Encryption Standard* (AES) 256.

```
APP_KEY=base64:T0u0zxSBwk1AX3TkAwnD5roVLqLASbXSQk7/5HozAYA=
```

Gambar 16. *Advanced Encryption Standard* (AES) 256 *Key*.

Implementasi AES-256 pada enkripsi pin pengguna menggunakan kunci sepanjang 256-bit untuk mengenkripsi dan mendekripsi data dalam blok 128-bit. Proses enkripsi terdiri dari 14 putaran yang mencakup operasi *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey*. Setiap putaran menggunakan sub-kunci yang dihasilkan melalui *Key Expansion*. Dekripsi mengikuti langkah yang sama namun dengan urutan operasi yang dibalik. AES-256 dapat digunakan dalam berbagai mode operasi seperti ECB, CBC, dan CTR untuk meningkatkan keamanan. Implementasi yang benar memerlukan perhatian pada pengelolaan kunci dan pemilihan mode yang sesuai.

E-ISSN: 2985-3532

```php
<?php

namespace App\Http\Controllers\Auth;

use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\VerifiesEmails;
use App\OtpCode;
use App\User;
use App\Mail\OtpMail;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Mail;
use Carbon\Carbon;

class VerificationController extends Controller
{
    public function showVerifyOtp()
    {
        return view('auth.verify');
    }

    public function verify(Request $request)
    {
        $request->validate([
            'email' => 'required|email',
            'otp' => 'required|string|size:6',
        ]);

        $user = User::where('email', $request->email)->first();

        if (!$user) {
            return back()->withErrors(['email' => 'Email tidak ditemukan.']);
        }

        $otpCode = OtpCode::where('user_id', $user->id)
            ->where('code', $request->otp)
            ->where('expire_at', '>', Carbon::now())
            ->first();

        if (!$otpCode) {
            return back()->withErrors(['otp' => 'Kode OTP tidak valid atau sudah kadaluarsa.']);
        }

        $user->update(['is_verified' => true]);
        $otpCode->delete();

        auth()->login($user);
        return redirect('/set-pin')->with('success', 'Akun berhasil diverifikasi.', 'Silahkan Buat Pin Anda!');
    }

    public function resend(Request $request)
    {
        $request->validate(['email' => 'required|email']);

        $user = User::where('email', $request->email)->first();

        if (!$user) {
            return back()->withErrors(['email' => 'Email tidak ditemukan.']);
        }

        // Hapus OTP lama jika ada
        OtpCode::where('user_id', $user->id)->delete();

        // Buat OTP baru
        $otp = str_pad(random_int(0, 999999), 6, '0', STR_PAD_LEFT);

        OtpCode::create([
            'user_id' => $user->id,
            'code' => $otp,
            'expire_at' => Carbon::now()->addMinutes(5),
        ]);

        Mail::to($user->email)->send(new OtpMail($otp, $user->name));

        return back()->with('message', 'Kode OTP baru telah dikirim.');
    }
```

Gambar 17. Implementasi *One-Time Password* (OTP).

Mekanisme One-Time Password (OTP) pada kode ini dirancang untuk meningkatkan keamanan dalam proses otentikasi sistem berbasis Laravel. OTP dihasilkan secara dinamis melalui fungsi *resend()* ketika pengguna mengajukan permintaan pengiriman ulang kode. Proses ini diawali dengan validasi email pengguna, di mana sistem akan mencari data pengguna pada basis data untuk memastikan keabsahannya. Selain itu, sistem juga menghapus kode OTP lama yang mungkin masih aktif untuk menjamin bahwa hanya ada satu kode OTP yang valid pada suatu waktu.

Selanjutnya, sistem menghasilkan kode OTP berupa angka enam digit, menyimpan kode tersebut ke dalam tabel *OtpCode* dengan masa berlaku selama lima menit, dan mengirimkannya ke alamat email pengguna menggunakan kelas *OtpMail*. Dengan langkah ini, OTP yang dikirimkan bersifat unik dan hanya berlaku untuk sementara waktu, sehingga meningkatkan keamanan sistem.

Proses verifikasi OTP dilakukan melalui fungsi *verify()*, yang bertugas memvalidasi masukan berupa email dan kode OTP. Setelah validasi, sistem mencocokkan OTP yang dimasukkan dengan data yang tersimpan dalam tabel *OtpCode*. Verifikasi dilakukan untuk memastikan bahwa kode sesuai dengan pengguna terkait, belum kedaluwarsa, dan masih aktif. Jika validasi berhasil, sistem akan memperbarui status pengguna menjadi terverifikasi, menghapus kode OTP dari basis data untuk mencegah penggunaan ulang, dan mengarahkan pengguna ke halaman pembuatan PIN.

Namun, jika kode OTP tidak valid atau telah kedaluwarsa, sistem akan menampilkan pesan kesalahan kepada pengguna. Dengan penerapan mekanisme ini, sistem dapat memberikan perlindungan yang lebih baik terhadap akses tidak sah, sekaligus menjamin kelancaran proses autentikasi pengguna.

2.3. Kendala Dalam Implementasi

Pada implementasi Metode Hashing Blowfish terdapat beberapa kendala, seperti ukuran blok 64-bit yang lebih kecil dibandingkan algoritma modern, meningkatkan risiko collision saat mengenkripsi data besar. Kecepatan enkripsi juga kurang optimal pada sistem modern, karena banyaknya putaran yang diperlukan. Selain itu, penggunaan kunci yang lebih pendek (misalnya 32 bit) dapat meningkatkan risiko serangan *brute-force*.

Blowfish juga kurang cocok untuk mengenkripsi banyak data dengan kunci yang sama dalam waktu lama, yang berisiko mengungkapkan pola enkripsi. Implementasinya juga lebih kompleks, karena kesalahan dalam pengelolaan *P-array* dan *S-boxes* dapat membuka celah keamanan.

Sedangkan untuk Algoritma AES 256, Implementasi terdapat kendala, seperti kinerja yang lebih lambat pada perangkat dengan sumber daya terbatas karena penggunaan kunci 256-bit dan 14 putaran enkripsi. Algoritma ini juga memerlukan memori lebih besar untuk menyimpan kunci dan tabel, serta implementasi yang lebih kompleks.

AES-256 rentan terhadap serangan side-channel jika tidak diterapkan dengan hati-hati, dan penggunaan kunci 256-bit seringkali menambah overhead tanpa memberikan peningkatan keamanan yang signifikan dibandingkan AES-128. Oleh karena itu, meskipun aman, AES-256 bisa kurang efisien di beberapa aplikasi dengan keterbatasan sumber daya.

4. PENGUJIAN WEBSITE

Salah satu pengujian yang dapat digunakan untuk menguji sistem adalah metode *blackbox testing*, yang berfokus pada pengujian fungsionalitas. Pengujian kotak hitam memungkinkan pengembang memastikan bahwa sistem yang dirancang memiliki kemudahan penggunaan dan responsivitas yang baik. Dengan demikian, metode ini memastikan bahwa sistem yang dikembangkan sesuai dengan persyaratan fungsional, memenuhi kebutuhan

E-ISSN: 2985-3532

serta spesifikasi yang ditetapkan, dan mampu memenuhi ekspektasi pengguna. Sistem dianggap berhasil melewati pengujian jika outputnya sesuai dengan yang diharapkan. Sebaliknya, apabila outputnya tidak sesuai, sistem dinyatakan gagal dalam pengujian.

Tabel 1. Pengujian *Multi Factor Authentication* Pada *E-Commerce* dengan Metode *Blackbox Testing.*

No	Pengujian	*Test Case*	Hasil Didapat	Hasil Pengujian	Keterangan
1	Menu *register*	Melakukan registrasi	Sistem berhasil menyimpan data registrasi *user*	Sesuai Harapan	Valid
2	Menu *login*	Melakukan *login*	Sistem berhasil menerima *request login* dan membuat kode OTP yang dikirimkan pada alamat email *user* yang telah didaftarkan saat registrasi	Sesuai Harapan	Valid
3	Menu verifikasi OTP	Melakukan verifikasi kode OTP	Sistem berhasil menerima *request* verifikasi kode OTP jika kode yang dimasukkan valid	Sesuai Harapan	Valid
4	Menu Pembuatan PIN	Membuat PIN	Sistem berhasil menerima *request* pembuatan PIN	Sesuai Harapan	Valid

IV. KESIMPULAN

Penerapan metode keamanan berbasis *Multi-Factor Authentication* (MFA) dengan kombinasi *hashing* menggunakan algoritma *Blowfish*, enkripsi menggunakan algoritma *Advanced Encryption Standard* (AES) 256, serta mekanisme *One-Time Password* (OTP) terbukti memberikan perlindungan yang signifikan terhadap data dan transaksi dalam sistem e-commerce. Algoritma *Blowfish* efektif dalam mengamankan password melalui proses *hashing*, sementara AES 256 mampu mengenkripsi PIN dengan tingkat keamanan yang sangat tinggi.

Penggunaan OTP sebagai lapisan tambahan autentikasi juga meningkatkan perlindungan terhadap akses tidak sah. Dengan implementasi yang tepat, sistem keamanan ini dapat melindungi pengguna dari berbagai ancaman siber, seperti peretasan dan pencurian data, sekaligus mendukung perkembangan *e-commerce* yang lebih aman dan terpercaya.

E-ISSN: 2985-3532

V. UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada semua orang yang telah membantu. Tulisan ini tidak akan menjadi kenyataan jika mereka tidak mendukung dan berpartisipasi. Penulis mengucapkan terima kasih kepada Allah SWT, orang tua dan keluarga, tim peneliti, teman-teman yang membantu dan dosen pembimbing serta dirinya sendiri.

VI. REFERENSI

[3] Kehista, A. P., Fauzi, A., Tamara, A., Putri, I., Fauziah, N. A., Klarissa, S., & Damayanti, V. B. (2023). Analisis Keamanan Data Pribadi pada Pengguna E-Commerce: Ancaman, Risiko, Strategi Keamanan (Literature Review). Jurnal Ilmu Manajemen Terapan, 4(5), 625-632.

[4] Putra, S. Z., Harianto, S. T., & Matondang, Y. C. (2023). Analisis Pengaruh E-Commerce: Studi Literatur Terhadap Pertumbuhan Ekonomi UMKM. Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer, 3(2), 119-131.

[5] Ramadhani, N., & Nasution, M. I. P. (2024). Tantangan Dan Solusi Keamanan Siber Dalam Transaksi E-Commerce. Jurnal Penelitian Sistem Informasi (JPSI), 2(2), 134-144.

[6] Azhari, M., Mulyana, D. I., Perwitosari, F. J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). Jurnal Pendidikan Sains dan Komputer, 2(01), 163-171.

[7] Karsana, I. W. W., & Mahendra, G. S. (2021). Sistem Informasi Geografis Pemetaan Lokasi Puskesmas Menggunakan Google Maps API di Kabupaten Badung. Jurnal Komputer Dan Informatika, 9(2), 160-167.

[8] Fachri, B., & Surbakti, R. W. (2021). Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis Website (Studi Kasus: Asco Jaya). Journal Of Science And Social Research, 4(3), 263-267

[9] Vicky, V. O., & Syaripudin, A. (2022). Perancangan Sistem Informasi Absensi Pegawai Berbasis Web Dengan Metode Waterfall (Studi Kasus: Kantor Dbpr Tangerang Selatan). OKTAL: Jurnal Ilmu Komputer dan Sains, 1(01), 17-26.