OTENTIKASI MULTI-FAKTOR

By . Ludang prasetyo .N

UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA

22 september 2024

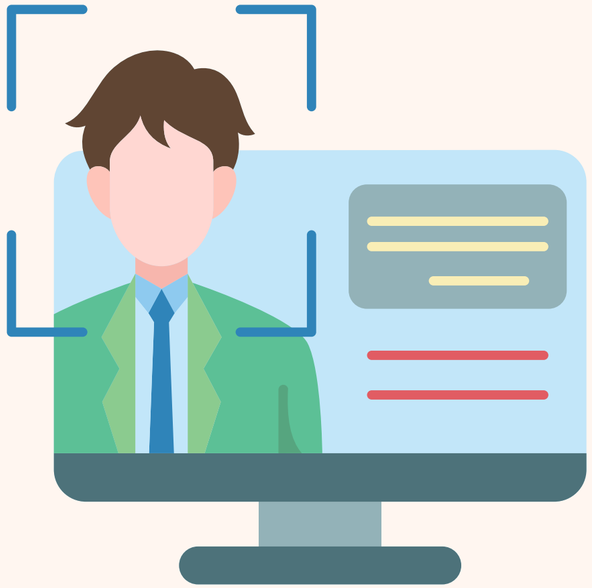
Nim 225510017

utdi .ac.id

Apa itu ...?

Otentikasi Multi-Faktor (MFA)

Ini adalah suatu metode verivikasi keamanan yang menggunakan lebih dari satu cara untuk memverivikasi identitas seseorang Untuk memberikan keamanan ekstra selain menggunakan kata sandi yang rentan untuk di bobol

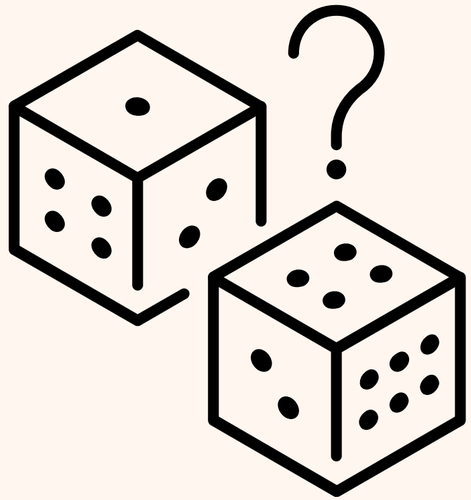


Dasar teorinya

Otentifikasi ini berfokus pada peningkatan keamanan dengan mengkombinasikan lebih dari satu metode verifikasi identitas untuk memastikan bahwa hanya pengguna yang sah yang bisa mendapatkan akses ke sistem

1. **Konsep Faktor Otentikasi MFA menggunakan beberapa faktor otentikasi, yang biasanya dikelompokkan menjadi tiga kategori utama**

   - **Sesuatu yang Anda Ketahui (Something you know): Misalnya, password, PIN, atau jawaban atas pertanyaan keamanan.**
   - **Sesuatu yang Anda Miliki (Something you have): Misalnya, perangkat fisik seperti smartphone, kartu kredit, atau token.**
   - **Sesuatu yang Anda Ada (Something you are): Misalnya, biometrik seperti sidik jari, pemindaian wajah, atau retina mata.**



2 Prinsip Keamanan Berlapis

MFA memanfaatkan lebih dari satu lapisan keamanan untuk memastikan bahwa meskipun satu lapisan (misalnya, password) berhasil dibobol, lapisan lain (misalnya, OTP atau autentikasi biometrik) masih melindungi sistem dari akses yang tidak sah.

3 Kerentanan Faktor Tunggal

Sistem otentikasi berbasis satu faktor (misalnya, hanya password) lebih rentan terhadap berbagai serangan seperti phishing, brute force, atau credential stuffing. MFA mengurangi risiko ini dengan memaksa penyerang untuk melewati beberapa lapisan keamanan yang berbeda, yang lebih sulit diatasi.

4 Keamanan Berbasis Risiko

sering diintegrasikan dengan sistem keamanan berbasis risiko, di mana tingkat otentikasi yang diperlukan bisa bervariasi tergantung pada risiko yang terdeteksi, seperti login dari lokasi atau perangkat yang mencurigakan.

Penerapan Otientikasi

1. **Banking Online**

   Ketika kamu ingin login ke akun perbankan online, langkah pertama adalah memasukkan username dan password. Setelah itu, kamu akan menerima kode OTP (One-Time Password) yang dikirim melalui SMS atau aplikasi autentikasi untuk memasukkan kode tersebut sebelum akses diberikan.

2. **Email**

   Saat login ke akun email, setelah memasukkan password, kamu diminta untuk memverifikasi identitas dengan menggunakan aplikasi autentikasi seperti Google Authenticator atau dengan menggunakan fingerprint atau face recognition di smartphone.

3. **Aplikasi Cloud (misalnya Google Drive atau Dropbox)**

   Kamu masuk menggunakan password, tetapi kemudian diminta untuk memasukkan kode verifikasi yang dikirim ke email atau melalui aplikasi autentikasi sebagai lapisan keamanan tambahan.

4. **E-commerce (misalnya Tokopedia, Shopee)**

   Saat melakukan login atau pembayaran, setelah memasukkan username dan password, pengguna diminta untuk memasukkan kode OTP yang dikirimkan melalui SMS atau email sebagai lapisan keamanan tambahan. Beberapa platform juga menyediakan opsi verifikasi melalui aplikasi autentikasi atau biometrik seperti sidik jari di smartphone.

Cara kerjanya

Otentikasi Multi-Faktor (MFA) dengan meminta pengguna untuk melewati beberapa tahap verifikasi identitas sebelum diberikan akses ke suatu sistem atau layanan.

Tahapan veririvikasi

1. **Pengguna Memasukkan Kredensial Pertama**

   Pengguna pertama-tama memasukkan informasi yang biasa digunakan untuk login, seperti username dan kata sandi.

❷ Verifikasi Faktor Kedua

Ini biasanya setelah memasukan sandi biasanya user di suruh memasukan code “ OTP ” Yang akan di kirimkan melalui Nomor hp & Email

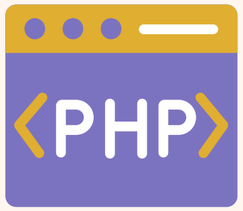
Biometrik, seperti sidik jari atau pengenalan wajah. Ini adalah contoh sesuatu yang kamu miliki secara biologis.

❸ Akses Diberikan

Saat pengguna / user sudah memberrikan verivikasi (OTP ,Biometrik, seperti sidik jari atau pengenalan wajah.) Maka akan langsung di berikan akses untuk masuk

Tehnik Peorograman

Dalam pemrograman otientivikasi Otentikasi Multi-Faktor (MFA) Biasanya menggunakan Bahasa pemrograman Node.JS dan Google Authenticator sebagai aplikasi otentikasi berbasis waktu (Time-Based OTP)



Php

PhpMailer

Penerapan dalam Programan

Langkah- langkah dasar Implementasi Otentikasi Multi-Faktor(MFA) untuk memverivikasi code OTP Untuk Dari email untuk login/Masuk kedalam web,instansi pemerintah,aplikasi,Dll

Kalian juga membutuhkan librari yang harus terpasang untuk menyediakan cara untuk mengirim email dari server.

API !

API dibuat secara langsung menggunakan PhpMailer dengan Main()

Librari !

Kalian bisa install librari yang di butuhkan seperti (PhpMailler)

ALGORITMA / ALIR

- Mulai
- Pengguna diarahkan ke Halaman Pendaftaran.
- Pengguna memasukkan Email dan Password.
- Simpan data ke Database.
- Pengguna diarahkan ke Halaman Login.
- Pengguna memasukkan Email dan Password.
- Sistem melakukan Cek Database untuk validasi.

  Jika Email/Password Valid, lanjut ke langkah berikutnya.

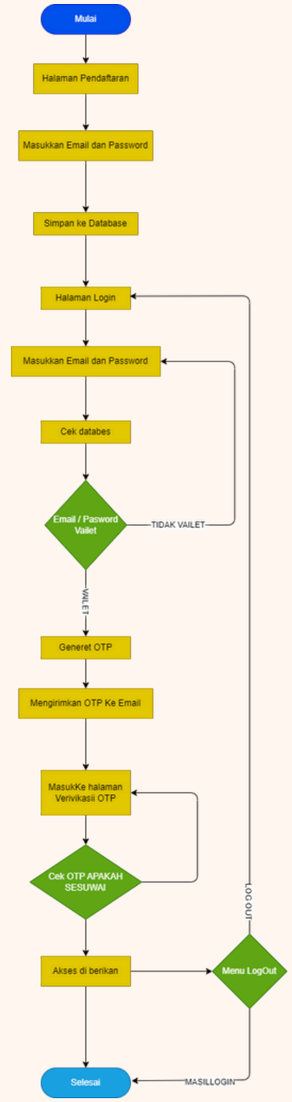
  Jika Email/Password Tidak Valid, kembali ke halaman login.

- Generate OTP.
- Mengirimkan OTP ke Email pengguna.
- Pengguna diarahkan ke Halaman Verifikasi OTP.
- Pengguna memasukkan OTP.
- Sistem melakukan Cek OTP.

  Jika OTP Sesuai, akses diberikan dan pengguna diarahkan ke halaman log-out.

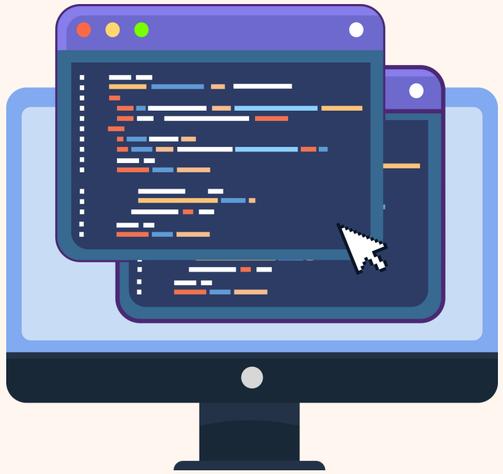
  Jika OTP Tidak Sesuai, kembali ke halaman verifikasi OTP.

- Bila pengguna mau logout langsung kembali ke menu login
- Selesai



Copy.nugra.online

IMPLEMENTASI CODE



1 Instal Librari Php Di command prom / CMD

```
composer require phpmailer/phpmailer
```

Ini akan menambahkan fils phpMiler atau folder Vendor

```
session_start();
include 'config.php';
require 'vendor/autoload.php'; // Pastikan PHPMailer terinstall
```

Code untuk memasukan Librari / Memanggil librari

2 **Menambahkan emailyang akan mengirimkan otp**

```
// Kirim email dengan OTP
$mail = new PHPMailer\PHPMailer\PHPMailer();
$mail->isSMTP();
$mail->Host = 'smtp.gmail.com'; // SMTP server
$mail->SMTPAuth = true; // Mengaktifkan autentikasi SMTP
$mail->Username = 'nugra315@gmail.com'; // Email Anda
$mail->Password = PASWORD AKUN ADA // Password email Anda
$mail->SMTPSecure = 'tls'; // Gunakan 'ssl' jika menggunakan port 465
$mail->Port = 587; // Port untuk TLS
```

Jadi kalian juga membutuhkan email dami / kusus untuk mengirimkan otpke rmail user yang login

3 Kode untuk membuat Otpsecara acak

```
// Mengirim OTP
$otp = rand(100000, 999999);
$_SESSION['otp'] = $otp;
```

Ini Otp Akan di buat secara acak

```
$mail->setFrom('Ludang Prasetyo Nugroho', 'Nugra21'); // Format pengirim
$mail->addAddress($email);

$mail->isHTML(true);
$mail->Subject = 'Code OTP N21.WERE';
$mail->Body = "Code verivikasi anda adalah: $otp";
```

Dan ini adalah Format pengiriman Otp ke Email User

4 Mencocokan Otp

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $otpInput = $_POST['otp'];

  if ($otpInput == $_SESSION['otp']) {
    // Verifikasi berhasil
    header("Location: Dasbord\index.php");
    exit();
  } else {
    $error_message = "OTP tidak valid!";
  }
}
```

Ini code untuk mencocokan apakah OTP sama seperti yang di kirimkan di email User

5 DB coneksi ke databes

```
$servername = "localhost";
$username = "root"; // Ganti jika menggunakan username berbeda
$password = ""; // Ganti jika menggunakan password
$dbname = "your_database"; // Ganti dengan nama database Anda

// Membuat koneksi
$conn = new mysqli($servername, $username, $password, $dbname);

// Cek koneksi
if ($conn->connect_error) {
  die("Koneksi gagal: " . $conn->connect_error);
}
```

Code untuk menyambungkan ke databes di PHPMyadmin

6 Saat ada user baru yang registrasi

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $email = $_POST['email'];
  $password = password_hash($_POST['password'], PASSWORD_DEFAULT);

  // Simpan ke database
  $sql = "INSERT INTO users (email, password) VALUES ('$email', '$password')";

  if ($conn->query($sql) === TRUE) {
    $message = "Pendaftaran berhasil! Silakan login.";
  } else {
    $message = "Error: " . $conn->error;
  }
}
```

Code untuk User yang baru mendaftar maka dana akan di simpan di databes

7 Code untuk dasbord

```
<?php
session_start();
if (!isset($_SESSION['otp'])) {
  header("Location: ../login.php");
  exit();
}
?>
```

Code untuk menutup jadi saat user belum login maka tidak bisa mesuk ke dalam dasbord

8 LogOut

```
<?php
session_start();
session_destroy(); // Hapus semua session
header("Location: login.php"); // Redirect ke halaman login
exit();
?>
```

Code untuk LogOut saat user akanlogout maka kode ini akan di jalankan

KELEBIHAN

Keamanan yang Ditingkatkan

- **MFA menambah lapisan perlindungan ekstra. Bahkan jika kata sandi bocor atau dicuri, penyerang masih memerlukan faktor tambahan (seperti OTP atau biometrik) untuk mengakses akun.**

Mengurangi Risiko Pembobolan

- **Dengan menerapkan MFA, risiko akses tidak sah ke akun berkurang secara signifikan, yang penting untuk melindungi data sensitif, terutama pada aplikasi perbankan atau layanan kesehatan.**

Meningkatkan Kepercayaan Pengguna

- **Pengguna cenderung merasa lebih aman menggunakan layanan yang menerapkan MFA, karena mereka tahu ada langkah tambahan untuk melindungi informasi pribadi mereka.**

KEKURANGAN

Ketersediaan Teknologi

- **Tidak semua pengguna memiliki akses yang konsisten ke perangkat atau aplikasi yang diperlukan untuk otentikasi, seperti smartphone untuk menerima OTP atau aplikasi lainya.**

Risiko Kehilangan Akses

- **Jika pengguna kehilangan perangkat yang digunakan untuk otentikasi (misalnya, ponsel untuk menerima OTP), mereka mungkin tidak bisa mengakses akun mereka. Proses pemulihan akses bisa rumit dan memakan waktu.**

Biaya Implementasi

- **Untuk organisasi, menerapkan MFA bisa memerlukan investasi dalam perangkat keras, perangkat lunak, atau infrastruktur yang mungkin tidak terjangkau untuk semua perusahaan, terutama bisnis kecil.**

CONTOH LYANAN YANG MEMAKAI (MFA)

- Google (Gmail)
- Microsoft (Akun Microsoft)
- Facebook
- Twitter
- Amazon Web Services (AWS)
- Bank Mandiri
- Bank BCA
- Tokopedia
- Slack
- Dropbox
- Salesforce
- PayPal

KESIMPULAN

MFA adalah langkah penting untuk meningkatkan keamanan, tetapi harus diimbangi dengan kemudahan penggunaan dan aksesibilitas. Organisasi perlu mempertimbangkan baik keuntungan maupun kerugian saat merancang sistem otentikasi mereka agar efektif dan nyaman bagi pengguna.

Daftar pustaka

https://www.fraud.com/post/multi-factor-authentication

https://stytch.com/blog/how-to-enforce-multi-factor-authentication-with-node-js/

SILAHKAN BERTANYA