NAMA : RR ANINDYA RAHAYULIANTI SARI

NIM : 201011401466

MATA KULIAH : PEMPROGRAMAN WEB 2

SOAL DAN JAWABAN

1. Anda diminta untuk membangun sebuah sistem autentikasi pengguna pada sebuah aplikasi web. Sistem ini harus memiliki fitur lupa password dan perlindungan terhadap serangan brute force. Pertanyaan:

a) Algoritma hashing apa yang paling cocok digunakan untuk menyimpan password pengguna? Jelaskan alasannya dan berikan contoh programnya.

Jawab : Algoritma hashing yang paling cocok untuk menyimpan password pengguna adalah algoritma bcrypt, Argon2, atau PBKDF2. Algoritma-algoritma ini dirancang khusus untuk hashing password dan memiliki fitur penting seperti *salt*, *key stretching*, dan kontrol terhadap tingkat kesulitan.

Argon2:

- Alasan: Merupakan algoritma terbaru dan dianggap sebagai salah satu yang paling aman saat ini. Argon2 dirancang secara khusus untuk resistensi terhadap serangan dengan hardware khusus (ASIC) dan parallel computing.
- Keunggulan: Fleksibel, dapat disesuaikan dengan kebutuhan keamanan, dan memiliki parameter yang memungkinkan untuk menyesuaikan tingkat kesulitan hashing.

bcrypt:

- Alasan: Telah terbukti aman selama bertahun-tahun dan banyak digunakan dalam berbagai aplikasi. bcrypt dirancang untuk lambat secara sengaja, sehingga membuat serangan brute-force menjadi lebih sulit.
- Keunggulan: Sederhana, mudah diimplementasikan, dan memiliki parameter untuk menyesuaikan tingkat kesulitan hashing.

PBKDF2:

- Alasan: Algoritma yang kuat dan telah teruji waktu. PBKDF2 menggunakan fungsi hash iteratif untuk meningkatkan keamanan.
- Keunggulan: Fleksibel, dapat menggunakan berbagai fungsi hash sebagai dasar.

```
<?php

// Fungsi untuk hash password

function hashPassword($password) {

   return password_hash($password, PASSWORD_BCRYPT);

}

// Fungsi untuk memverifikasi password

function verifyPassword($password, $hash) {
```

```
    return password_verify($password, $hash);
}
// Contoh penggunaan
$hashedPassword = hashPassword("password123");
if (verifyPassword("password123", $hashedPassword)) {
    echo "Password benar";
} else {
    echo "Password salah";
}
```

b) Bagaimana cara Anda mengimplementasikan fitur lupa password dengan aman? Buatkan contoh programnya!

Jawab : Implementasi fitur lupa password harus dirancang dengan sangat hati-hati agar tetap aman. Password lama tidak boleh disimpan atau dikirim kembali ke pengguna.

Fitur lupa password adalah fitur penting dalam sebuah aplikasi web untuk membantu pengguna yang lupa kata sandi mereka. Implementasi yang baik akan memastikan keamanan data pengguna dan kemudahan akses bagi pengguna yang memang membutuhkannya.

```
// Controller
public function forgotPassword(Request $request)
{
    $user = User::where('email', $request->email)->first();
    if (!$user) {
        return back()->withErrors(['email' => 'Email tidak ditemukan']);
    }
    $token = Str::random(60);
    DB::table('password_resets')->insert([
        'email' => $user->email,
        'token' => $token,
        'created_at' => now()
    ]);
    // Kirim email dengan link reset password yang mengandung token
    Mail::to($user->email)->send(new ForgotPasswordMail($token));
    return back()->with('message', 'Link reset password telah dikirim ke email Anda.');
```

```
}
public function resetPassword(Request $request)
{
    $passwordReset = DB::table('password_resets')->where([
        'email' => $request->email,
        'token' => $request->token
    ])->first();
    if (!$passwordReset) {
        // Token tidak valid atau sudah kadaluarsa
        return back()->withErrors(['token' => 'Token tidak valid']);
    }
    $user = User::where('email', $request->email)->first();
    $user->password = Hash::make($request->password);
    $user->save();
    DB::table('password_resets')->where('email', $request->email)->delete();
    return redirect('/login')->with('message', 'Password berhasil diubah');
}
```

c) Teknik apa saja yang dapat Anda gunakan untuk mencegah serangan brute force? Berikan contoh implementasi programnya!

Jawab : Untuk mencegah serangan *brute force*, ada beberapa teknik yang dapat diterapkan. Teknik-teknik ini dirancang untuk memperlambat atau memblokir serangan yang mencoba menebak password dengan mencoba banyak kombinasi.

Serangan brute force adalah upaya sistematis untuk menebak kata sandi dengan mencoba semua kemungkinan kombinasi karakter. Berikut beberapa teknik yang dapat Anda gunakan untuk mencegah serangan ini:

1. Pembatasan Percobaan Login

- Ide: Batasi jumlah percobaan login yang gagal dalam waktu tertentu.
- Implementasi (PHP, Laravel):

```
public function login(Request $request)
{
    $credentials = $request->validate([
        'email' => ['required', 'email'],
        'password' => ['required'],
```

```
        ]);
        // Cek apakah pengguna sudah mencoba login terlalu banyak kali
        if (RateLimiter::tooManyAttempts($request, 5)) {
            return $this->fail('You are attempting to log in too many times.');
        }
        // ... logic untuk melakukan login ...
    }
```

- Laravel: Gunakan fitur RateLimiter untuk membatasi percobaan login.

2. CAPTCHA

- Ide: Gunakan CAPTCHA untuk membedakan antara manusia dan bot.
- Implementasi:
    - Integrasikan layanan CAPTCHA seperti reCAPTCHA dari Google ke dalam form login.
    - Pastikan pengguna menyelesaikan CAPTCHA sebelum melanjutkan proses login.

3. Two-Factor Authentication (2FA)

- Ide: Tambahkan lapisan keamanan ekstra dengan meminta pengguna memasukkan kode verifikasi yang dikirim ke perangkat mereka.
- Implementasi:
    - Gunakan layanan 2FA seperti Authy atau Google Authenticator.
    - Kirim kode verifikasi melalui SMS, email, atau aplikasi autentikasi.

4. Delay pada Setiap Percobaan Login yang Gagal

- Ide: Tambahkan delay kecil pada setiap percobaan login yang gagal.
- Implementasi:
    - Gunakan fungsi sleep() (atau yang serupa) untuk menunda eksekusi setelah percobaan login gagal.

5. IP Blocking

- Ide: Blokir IP address yang melakukan terlalu banyak percobaan login gagal.
- Implementasi:
    - Catat IP address setiap kali ada percobaan login.
    - Jika IP address tertentu melebihi batas percobaan, blokir akses dari IP tersebut untuk sementara waktu.

6. Honey Pot

- Ide: Tambahkan field input palsu yang tidak terlihat untuk menjebak bot.

- Implementasi:
    - Tambahkan field input dengan nama yang tidak biasa dan sembunyikan dari pengguna.
    - Jika field ini terisi, kemungkinan besar itu adalah bot.

7. Password Policy yang Kuat

- Ide: Paksa pengguna untuk membuat password yang kuat dengan kombinasi huruf besar, huruf kecil, angka, dan karakter spesial.
- Implementasi:
    - Gunakan library atau fungsi bawaan bahasa pemrograman untuk memvalidasi kekuatan password.

8. Hashing Password yang Kuat

- Ide: Gunakan algoritma hashing yang kuat seperti bcrypt, Argon2, atau PBKDF2 untuk menyimpan password.
- Implementasi:
    - Lihat contoh implementasi pada jawaban sebelumnya.

2. Buatlah sebuah aplikasi web sederhana yang berfungsi sebagai to-do list. Aplikasi ini harus memiliki fitur untuk menambahkan, menghapus, dan mengedit tugas. Data tugas harus disimpan dalam database MySQL. Tugas:

a) Buatlah Query DDL untuk membuat database dan tabel yang akan digunakan untuk menyimpan data tugas!

Jawab :

b) Fungsi-fungsi PHP apa saja yang akan Anda gunakan untuk berinteraksi dengan database!

Jawab : Dalam PHP, ada berbagai fungsi bawaan yang dapat digunakan untuk berinteraksi dengan database. Pemilihan fungsi-fungsi tersebut bergantung pada jenis database yang digunakan.

Fungsi PHP yang digunakan untuk berinteraksi dengan database adalah:

- **mysql_connect**: dan mysqli_connect untuk membuat koneksi dari PHP ke server MySQL
- **mysql_select_db**: dan mysqli_select_db untuk memilih database yang akan digunakan
- **PDO (PHP Data Objects)**: interface universal yang disediakan PHP untuk berkomunikasi dengan database server

c) Buatlah REST API untuk GET, POST, PUT dan DELETE ke tabel todo_list!

Jawab :