

UAS TEORI SISTEM KEAMANAN DATA



Dosen Pembimbing : Muhammad A. Safi'ie

Disusun oleh :

Christo Gustawan Nugraha

V3422071 - TIC

MATA KULIAH TEORI SISTEM KEAMANAN DATA

SEKOLAH VOKASI

UNIVERSITAS SEBELAS MARET

Tahun Pelajaran 2023/2024

I. TUJUAN PRAKTIKUM

1. Keamanan data pengguna
2. Melindungi dari serangan Brute-Force
3. Kepatuhan dan Standar Keamanan
4. Peningkatan Keamanan Aplikasi
5. Pemahaman Konsep Keamanan

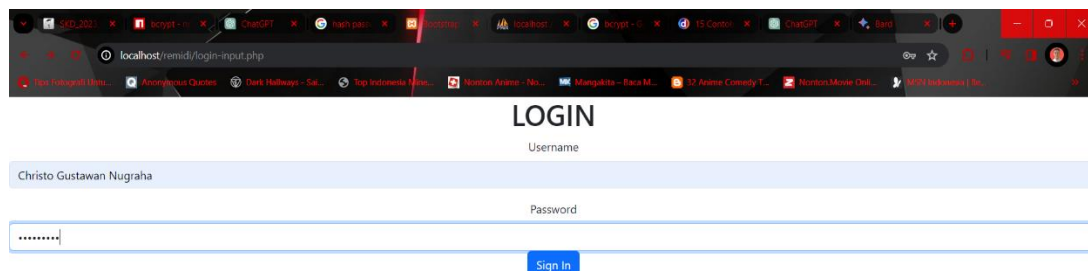
II. DASAR TEORI DAN KERJA

Bcrypt adalah teknik hash password yang aman. Ini menggunakan algoritma hash lambat, seperti Blowfish, serta salt untuk melindungi password pengguna. Prosesnya melibatkan penggabungan password dengan salt, diikuti oleh rotasi string menggunakan algoritma hash berulang kali berdasarkan nilai cost factor yang menentukan seberapa lambat proses tersebut.

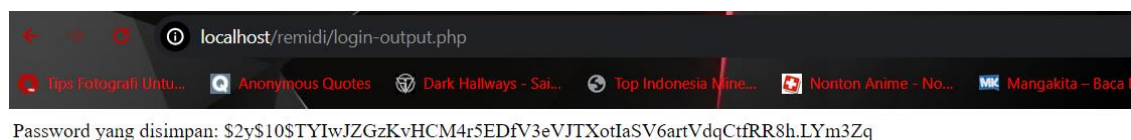
Keunggulan Bcrypt termasuk:

1. Keamanan: Dapat menahan serangan brute-force karena algoritma hash yang lambat.
2. Penggunaan Salt: Membuat hash password unik, sulit untuk memecahkan password menggunakan hash yang ada.
3. Kepatuhan: Memenuhi standar keamanan seperti OWASP Password Storage Cheat Sheet.
4. Kemudahan penggunaan: Tersedia dalam berbagai bahasa pemrograman dan mudah diimplementasikan.

Ini adalah solusi populer dan terpercaya untuk melindungi data sensitif pengguna pada aplikasi web dari serangan brute-force



```
login-output.php X
login-output.php
1  <html>
2
3  <body>
4      <?php
5          // Password yang ingin disimpan
6          $password = "password123";
7
8          // Buat hash dari password menggunakan bcrypt
9          $hashed_password = password_hash($password, PASSWORD_BCRYPT);
10
11         // Simpan $hashed_password di database atau tempat penyimpanan lainnya
12         // (biasanya disimpan bersamaan dengan informasi pengguna)
13         echo "Password yang disimpan: " . $hashed_password;
14     ?>
15
16
17 </body>
18
19 </html>
```

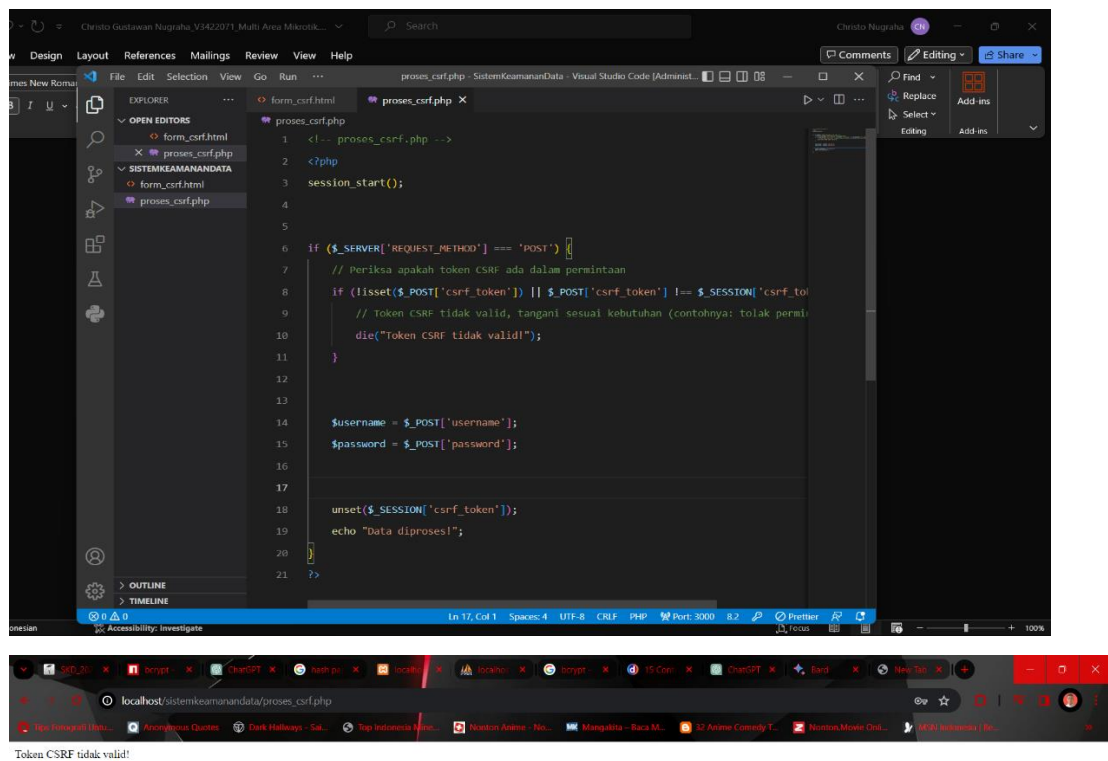


1. CSRF

CSRF (Cross-Site Request Forgery) adalah serangan keamanan di mana penyerang memanfaatkan kepercayaan yang diberikan kepada pengguna yang sudah diautentikasi untuk melakukan tindakan tertentu tanpa izin mereka. Dalam serangan ini, pengguna yang sudah diautentikasi secara tidak disengaja melakukan tindakan yang diminta oleh penyerang pada situs lain.

Contoh serangan CSRF: Seorang pengguna yang sudah login ke situs A mengunjungi situs yang dikendalikan oleh penyerang secara bersamaan. Situs tersebut bisa memicu permintaan otomatis ke situs bank menggunakan informasi otentikasi pengguna, melakukan transaksi tanpa izin mereka.

Pencegahan CSRF menggunakan token CSRF, nilai unik dalam formulir atau permintaan HTTP yang diperiksa oleh server. Token ini memverifikasi bahwa permintaan berasal dari situs web yang sah, bukan serangan luar. Dengan token CSRF, aplikasi dapat meningkatkan keamanan dan melindungi pengguna dari serangan yang memanfaatkan sesi autentikasi.

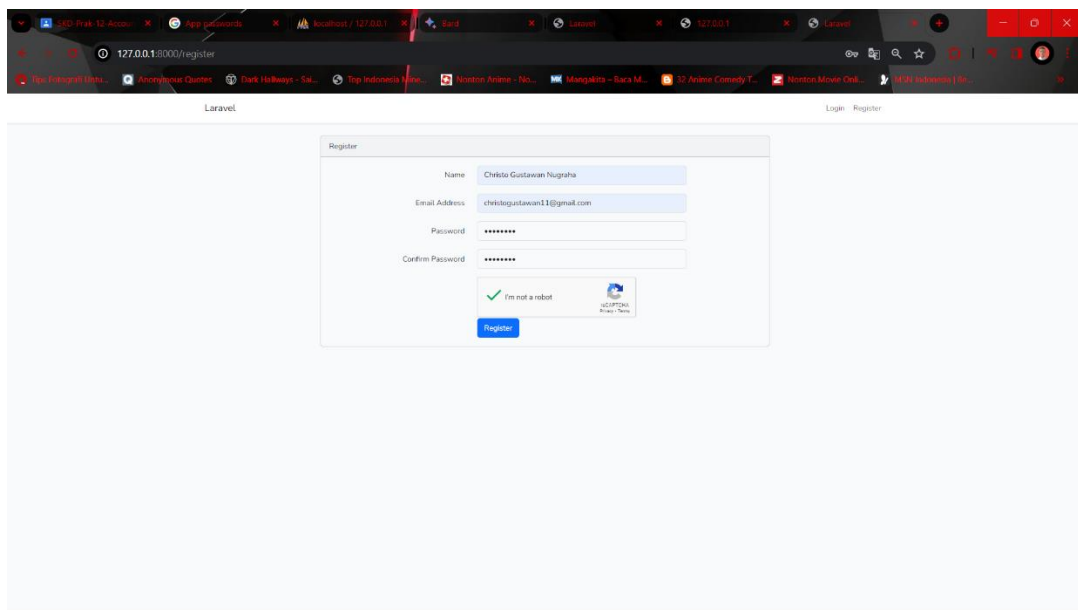


Ketika formulir dikirim, token CSRF yang disimpan di sesi dibandingkan dengan token yang dikirimkan bersama formulir dalam proses_csrf.php. Jika token tersebut tidak cocok atau tidak ada, maka permintaan akan dianggap tidak valid dan diproses sesuai kebutuhan (misalnya: tolak permintaan atau log pesan kesalahan).

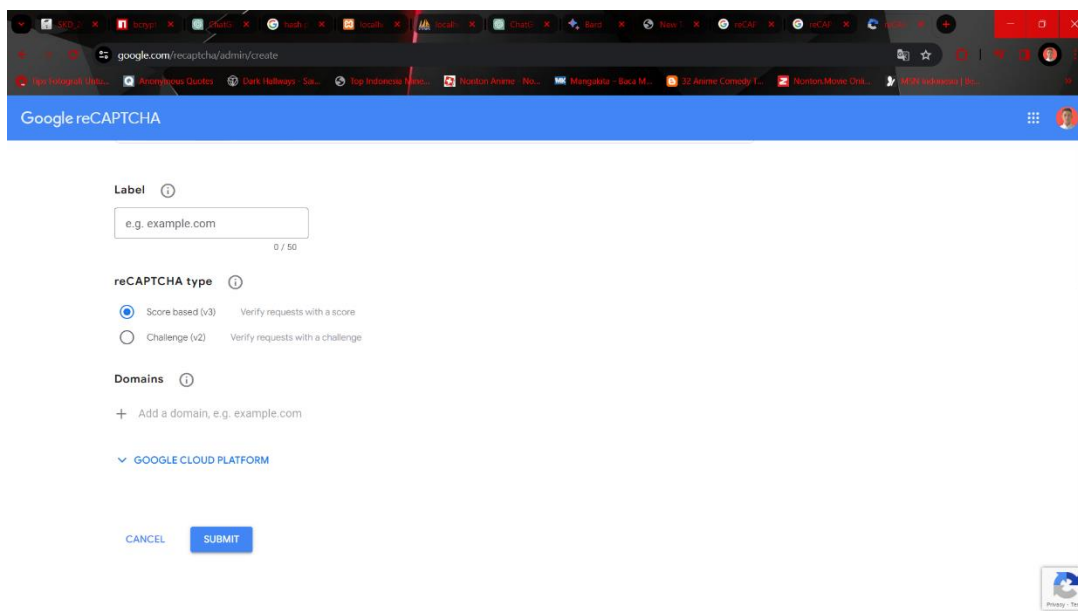
2. CAPTCHA

Captcha adalah metode untuk membedakan antara manusia dan mesin. Biasanya digunakan untuk mencegah spam dan serangan bot pada situs web dengan

menambahkan tugas tertentu pada form, seperti membedakan gambar, menuliskan teks dari gambar, atau menyelesaikan soal matematika sederhana. Jika pengguna menyelesaikan tugas tersebut, data dari form akan dikirimkan; jika tidak, data akan ditolak. Captcha membantu mencegah spam, serangan bot, dan meningkatkan keamanan situs web, terutama pada form login, komentar, dan pendaftaran.



Contoh Captcha menggunakan dari google reCaptcha V2



3. Multi-Aunthentication

Aktor-aktor dalam tugas proyek:

- Pengguna: Orang yang menggunakan aplikasi atau layanan.
- Admin: Bertanggung jawab untuk mengelola aplikasi atau layanan.
- Pengembangan: Bertanggung jawab untuk mengembangkan dan memelihara aplikasi atau

layanan.

- Operasi: Bertanggung jawab untuk mengoperasikan dan memelihara aplikasi atau layanan.

Multi-authentication:

- Menggunakan lebih dari satu metode untuk memverifikasi identitas pengguna.
- Metode autentikasi yang umum digunakan:

- Password
- Faktor kedua (2FA, otentikasi biometrik)

Cara menerapkan multi-authentication:

1. Identifikasi metode autentikasi yang akan digunakan.
2. Desain sistem multi-authentication.
3. Implementasi sistem multi-authentication.

Contoh penerapan multi-authentication:

- Aplikasi mobile: 2FA (password + kode OTP)
- Situs web: 2FA atau otentikasi biometrik

Manfaat multi-authentication:

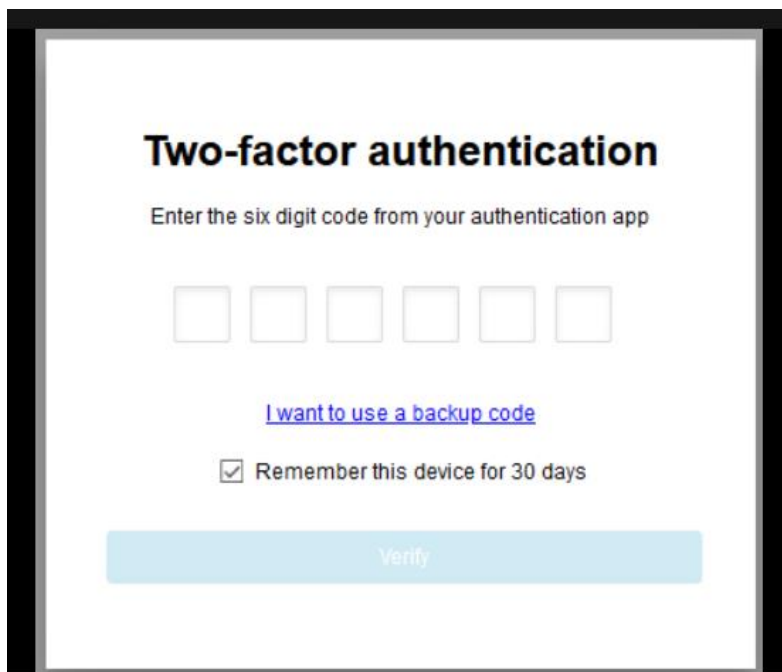
- Meningkatkan keamanan aplikasi atau layanan.

Keterbatasan multi-authentication:

- Dapat membuat proses autentikasi menjadi lebih rumit bagi pengguna.

Kesimpulan:

Multi-authentication adalah metode autentikasi yang dapat meningkatkan keamanan aplikasi atau layanan. Namun, penting untuk memilih metode autentikasi yang tepat dan mendesain sistem multi-authentication dengan baik agar tidak menyulitkan pengguna.

A screenshot of a web-based two-factor authentication interface. At the top, the title "Two-factor authentication" is displayed in bold black text. Below the title, a subtitle reads "Enter the six digit code from your authentication app". In the center, there are six empty square input boxes arranged horizontally for entering the code. Below these boxes is a blue hyperlink that says "I want to use a backup code". Underneath the link is a checkbox that is currently checked, followed by the text "Remember this device for 30 days". At the bottom of the form is a wide, light blue button with the word "Verify" in the center.

Contoh autentikasi dua faktor/ multi authentication

4. OTP

OTP adalah kode keamanan yang hanya dapat digunakan sekali untuk memverifikasi identitas pengguna. OTP digunakan sebagai metode autentikasi dua faktor (2FA) untuk meningkatkan keamanan akun atau layanan.

Manfaat OTP:

- Meningkatkan keamanan
- Meningkatkan kenyamanan
- Meningkatkan kepatuhan

Penggunaan OTP:

- Login
- Transaksi keuangan
- Akses perangkat

Cara menggunakan OTP:

1. Daftar untuk layanan yang menggunakan OTP.
2. Masukkan nomor telepon atau alamat email Anda.
3. Dapatkan kode OTP dari layanan tersebut.
4. Masukkan kode OTP ke dalam aplikasi atau layanan tersebut.

Tips menggunakan OTP:

- Simpan kode OTP di tempat yang aman.
- Jangan membagikan kode OTP dengan orang lain.
- Periksa dengan cermat kode OTP sebelum memasukkannya.



Contoh penggunaan OTP dari aplikasi Gojek

