Laporan Penanganan Kerentanan Keamanan pada Subdomain https://simak.darunnajah.ac.id/login (Dengan Masalah pada jQuery Versi Lama dan Pengelolaan Sesi Pengguna pada PHP)

---

1. Latar Belakang

Subdomain https://simak.darunnajah.ac.id/login merupakan aplikasi berbasis PHP yang menggunakan framework **CodeIgniter**. Saat dilakukan pengujian dengan memberikan input berupa URL yang berisi karakter khusus, yaitu tanda kutip tunggal ('), situs ini merespons dengan **PHP error** yang mengindikasikan adanya masalah dalam pengelolaan sesi pengguna.

Selain itu, Subdomain ini juga menggunakan versi lama dari **jQuery 1.2.1**, yang memiliki sejumlah kerentanannya sendiri, seperti potensi **Cross-Site Scripting (XSS)** dan **manipulasi DOM yang tidak aman**.

2. Deskripsi Masalah

Subdomain ini memberikan respons sebagai berikut setelah input diberikan:

A PHP Error was encountered
Severity: Notice

Message: Undefined property: CI_Exceptions::$session

Filename: html/error_general.php
Line Number: 68

Backtrace:
File: /srv/simak/application/views/errors/html/error_general.php
Line: 68
Function: _error_handler

File: /srv/simak/index.php
Line: 315
Function: require_once

An uncaught Exception was encountered
Type: Error

Message: Call to a member function userdata() on null

Filename: /srv/simak/application/views/errors/html/error_general.php

Line Number: 68

Backtrace:
File: /srv/simak/index.php
Line: 315
Function: require_once

3. Analisis Kerentanan

1. **Masalah dengan Session**: Pesan error mengindikasikan bahwa aplikasi tidak dapat mengakses properti `$session` pada objek `CI_Exceptions`. Hal ini menunjukkan bahwa sesi pengguna mungkin tidak diinisialisasi dengan benar atau tidak tersedia saat dibutuhkan.

2. **Fungsi `userdata()` pada Session yang Tidak Ada**: Pesan kedua menyebutkan bahwa aplikasi mencoba memanggil fungsi `userdata()` pada objek sesi yang tidak ada. Ini mengarah pada kesalahan dalam pengelolaan sesi atau pengaturan konfigurasi yang salah dalam aplikasi.

3. **Potensi Pengaruh Input Berbahaya**: Input berupa tanda kutip tunggal (`'`) dapat menyebabkan **SQL Injection**, **Cross-Site Scripting (XSS)**, atau mempengaruhi cara aplikasi memproses data. Ini menunjukkan bahwa aplikasi mungkin tidak sepenuhnya aman terhadap karakter-karakter berbahaya dalam inputan.

4. **Pengelolaan Error yang Tidak Tepat**: Pesan kesalahan yang sangat teknis ditampilkan kepada pengguna. Ini bisa mengekspos informasi sensitif mengenai arsitektur aplikasi dan potensi celah keamanan yang dapat dimanfaatkan oleh pihak yang tidak bertanggung jawab.

5. **Penggunaan jQuery Versi Lama (1.2.1)**:

    - **Cross-Site Scripting (XSS)**: jQuery versi lama seperti 1.2.1 tidak memiliki pembaruan untuk melindungi aplikasi terhadap teknik XSS modern. Penggunaannya dapat memungkinkan penyerang untuk menyuntikkan skrip berbahaya.

    - **Manipulasi DOM yang Tidak Aman**: Versi lama dari jQuery mungkin memiliki fungsi DOM yang rentan, seperti masalah dalam pemrosesan event atau perubahan data input pengguna yang dapat dieksploitasi oleh pihak ketiga.

    - **Kerentanan lainnya**: jQuery versi ini juga rawan terhadap berbagai jenis serangan yang sudah diperbaiki di versi-versi lebih baru.

4. Dampak

- **Keamanan Data Pengguna**: Jika masalah ini tidak diperbaiki, ada potensi kebocoran informasi pribadi pengguna karena masalah pengelolaan sesi dan input yang tidak difilter dengan benar.

- **Eksposur Informasi Aplikasi**: Menampilkan kesalahan teknis dapat memberikan penyerang wawasan lebih dalam tentang struktur aplikasi dan cara kerjanya, yang dapat digunakan untuk mengeksploitasi lebih lanjut.

- **Kebingungannya Pengguna**: Pengguna yang tidak familiar dengan pesan kesalahan ini bisa terjebak dalam pengalaman pengguna yang buruk, mengurangi kepercayaan terhadap website.

- **Risiko Keamanan Lainnya (dari jQuery Lama)**: Penggunaan jQuery versi lama berisiko tinggi karena celah keamanan yang diketahui yang bisa dimanfaatkan oleh penyerang untuk mengeksploitasi aplikasi.

5. Penyebab

1. **Kode PHP yang Tidak Aman**: Tidak adanya validasi atau sanitasi input yang memadai bisa menyebabkan terjadinya SQL injection, XSS, atau bahkan manipulasi session.

2. **Pengaturan Session yang Salah**: Pengaturan sesi yang salah atau tidak terinisialisasi dengan baik bisa menyebabkan kesalahan ketika aplikasi mencoba mengakses sesi untuk informasi pengguna.

3. **Penanganan Error yang Buruk**: Pengelolaan error yang tidak sesuai mengarah pada eksposur informasi teknis yang bisa digunakan untuk mengeksploitasi aplikasi.

4. **Penggunaan jQuery Versi Lama (1.2.1)**: jQuery versi lama memiliki beberapa kerentanannya yang sudah diketahui dan telah diperbaiki di versi yang lebih baru.

6. Saran Perbaikan

1. **Sanitasi Input**: Gunakan sanitasi dan validasi input secara ketat untuk menghindari manipulasi dengan karakter-karakter khusus (seperti tanda kutip) yang bisa menyebabkan celah keamanan.

    - Gunakan fungsi seperti `htmlspecialchars()`, `addslashes()`, atau yang setara untuk menghindari **SQL Injection** dan **XSS**.

2. **Perbaiki Pengelolaan Session**:
    - Pastikan bahwa session diinisialisasi dengan benar di seluruh bagian aplikasi sebelum digunakan.
    - Cek dan pastikan sesi pengguna sudah di-set sebelum fungsi seperti `userdata()` dipanggil untuk menghindari error.
3. **Penanganan Error yang Lebih Baik**:
    - Implementasikan **error handling** yang lebih aman, misalnya dengan menggunakan **try-catch** block untuk menangani exception dan menampilkan pesan error yang lebih umum kepada pengguna.
    - Jangan pernah menampilkan stack trace atau informasi teknis yang bisa digunakan untuk mengeksploitasi aplikasi.
4. **Perbarui dan Konfigurasi Framework dengan Benar**:
    - Pastikan bahwa semua framework dan dependensi aplikasi selalu diperbarui ke versi terbaru dan memiliki konfigurasi yang tepat untuk keamanan.
    - Periksa dan pastikan bahwa pengaturan sesi dan cache di CodeIgniter atau framework PHP lainnya sudah tepat dan tidak membuka celah.
5. **Update jQuery ke Versi Terbaru**:
    - Gantilah jQuery versi 1.2.1 yang usang dengan versi terbaru, seperti jQuery 3.x atau lebih tinggi, yang lebih aman dan memiliki fitur keamanan yang lebih baik untuk melindungi dari serangan XSS dan manipulasi DOM.
    - Pastikan aplikasi menggunakan **Content Security Policy (CSP)** dan **Subresource Integrity (SRI)** untuk memitigasi potensi eksploitasi dari penggunaan pustaka eksternal yang tidak aman.

7. Kesimpulan

Subdoimain menghadapi masalah serius yang dapat berdampak pada keamanan aplikasi dan pengalaman pengguna. Masalah pengelolaan sesi yang buruk, penanganan input yang tidak memadai, serta penggunaan jQuery versi lama memperburuk potensi kerentanannya. Disarankan untuk segera melakukan perbaikan terhadap pengelolaan sesi, sanitasi input, penanganan error, serta memperbarui jQuery ke versi yang lebih baru agar website menjadi lebih aman dan lebih stabil.