Nama :Amelia Nur Avivah

NIM : 22650169

Matkul : CyberCrime

**Penjelasan tentang Serangan Siber pada Aplikasi Web**

Serangan siber pada aplikasi web merupakan upaya oleh pihak yang tidak bertanggung jawab untuk mengeksploitasi kerentanannya dan merusak, mencuri, atau mengubah data yang ada dalam aplikasi web. Berikut adalah penjelasan detail mengenai beberapa jenis serangan siber yang sering terjadi pada aplikasi web:

1. SQL Injection

SQL Injection adalah jenis serangan yang memanfaatkan celah keamanan pada aplikasi web yang berhubungan dengan database. Pada serangan ini, penyerang menyisipkan query SQL berbahaya melalui input pengguna yang tidak divalidasi dengan baik. Hal ini bisa menyebabkan penyerang mengakses data sensitif dalam database, mengubah, atau bahkan menghapus data.

**Pencegahan:**

* Gunakan pernyataan yang diparameterkan atau prepared statements untuk mencegah eksekusi query yang berbahaya.
* Validasi dan filter input pengguna.
* Gunakan prinsip "least privilege" untuk hak akses ke database.

1. Cross-Site Scripting (XSS)

XSS adalah serangan di mana penyerang menyuntikkan skrip jahat ke dalam aplikasi web yang kemudian dijalankan oleh pengguna lain. Skrip ini biasanya dijalankan di sisi klien (browser pengguna) dan bisa mencuri informasi sensitif seperti cookie, token otentikasi, atau data lainnya.

**Jenis XSS:**

* Stored XSS: Skrip disimpan dalam database atau sumber daya lainnya dan dieksekusi setiap kali pengguna membuka halaman yang terpengaruh.
* Reflected XSS: Skrip disertakan dalam URL dan langsung dieksekusi oleh browser setelah pengguna mengklik link yang mengandung skrip tersebut.
* DOM-based XSS: Skrip jahat dieksekusi sebagai akibat dari manipulasi Document Object Model (DOM) di browser.

**Pencegahan:**

* Escape semua input yang diterima sebelum ditampilkan di halaman web.
* Gunakan header HTTP seperti `Content-Security-Policy (CSP)` untuk mencegah eksekusi skrip jahat.
* Validasi dan filter input pengguna.

1. Cross-Site Request Forgery (CSRF)

CSRF adalah jenis serangan di mana penyerang memanfaatkan kredensial pengguna yang sudah terautentikasi untuk melakukan aksi yang tidak diinginkan atas nama pengguna tersebut. Misalnya, mengirimkan permintaan untuk mengubah kata sandi atau melakukan transfer uang.

**Pencegahan:**

* Gunakan token anti-CSRF dalam setiap form dan permintaan HTTP untuk memverifikasi bahwa permintaan tersebut sah.
* Gunakan verifikasi metode permintaan yang lebih aman seperti `SameSite` cookies untuk membatasi pengiriman cookie dari domain yang berbeda.

1. Directory Traversal

Serangan Directory Traversal bertujuan untuk mengeksploitasi celah pada aplikasi web untuk mengakses file atau direktori yang berada di luar direktori yang diizinkan. Dengan teknik ini, penyerang bisa membaca file sistem yang sensitif seperti konfigurasi server atau file yang mengandung kredensial.

**Pencegahan:**

* Validasi dan sanitasi input pengguna untuk menghindari traversal direktori.
* Gunakan akses kontrol berbasis jalur yang ketat.

1. Denial of Service (DoS) dan Distributed Denial of Service (DDoS)

Serangan DoS atau DDoS bertujuan untuk membuat aplikasi web atau server menjadi tidak dapat diakses dengan cara membanjiri server dengan trafik yang sangat banyak. Pada serangan DDoS, penyerang menggunakan botnet (jaringan komputer yang telah terinfeksi) untuk melancarkan serangan dari berbagai sumber.

**Pencegahan:**

* Gunakan solusi perlindungan DDoS seperti penyaring trafik, firewall, atau layanan mitigasi DDoS.
* Terapkan rate limiting dan monitoring trafik untuk mendeteksi pola yang mencurigakan.

1. File Inclusion

Serangan File Inclusion terjadi ketika aplikasi web memungkinkan pengguna untuk memasukkan jalur file secara dinamis, tanpa validasi yang cukup. Penyerang bisa memanfaatkan ini untuk menyertakan file yang berbahaya dari server atau sistem lokal.

**Jenis:**

* Local File Inclusion (LFI): Menyertakan file dari server lokal.
* Remote File Inclusion (RFI): Menyertakan file dari sumber eksternal.

**Pencegahan:**

* Validasi dan sanitasi input file dengan ketat.
* Hindari penggunaan fungsi yang memungkinkan file yang tidak aman dimasukkan ke dalam aplikasi, seperti `include()` atau `require()` pada PHP.

1. Session Hijacking

Session Hijacking adalah serangan di mana penyerang mencuri sesi pengguna yang sah dan mengambil alih sesi tersebut. Hal ini sering dilakukan dengan mencuri cookie sesi yang disimpan di browser pengguna.

**Pencegahan:**

* Gunakan enkripsi untuk cookie dan sesi.
* Gunakan cookie dengan atribut `Secure` dan `HttpOnly` untuk meningkatkan keamanan.
* Terapkan timeout sesi yang singkat dan mekanisme pengakhiran sesi secara otomatis.

1. Insecure Deserialization

Insecure deserialization adalah serangan yang memanfaatkan celah dalam aplikasi yang menerima data yang tidak dipercaya (seperti objek serialized) dan kemudian mendeserialize data tersebut tanpa memverifikasi integritas atau keamanannya. Penyerang bisa menyisipkan data berbahaya yang dapat dieksekusi oleh aplikasi.

**Pencegahan:**

* Jangan deserialize data yang tidak dipercaya.
* Gunakan mekanisme verifikasi integritas seperti digital signatures atau checksum.

1. Broken Authentication

Serangan ini terjadi ketika aplikasi web gagal untuk melindungi kredensial pengguna atau mekanisme otentikasi, sehingga memungkinkan penyerang untuk memperoleh akses tanpa izin ke aplikasi.

**Pencegahan:**

* Gunakan otentikasi dua faktor (2FA).
* Implementasikan kebijakan kata sandi yang kuat dan aman.
* Pastikan kredensial disimpan dengan enkripsi yang kuat (misalnya bcrypt atau Argon2).

**Kesimpulan**

Serangan siber pada aplikasi web bisa sangat merugikan, baik dari segi kerugian data maupun reputasi. Oleh karena itu, sangat penting untuk membangun aplikasi dengan prinsip keamanan yang ketat, termasuk validasi input, penggunaan enkripsi, otentikasi yang kuat, dan perlindungan terhadap serangan yang umum seperti SQL Injection, XSS, dan CSRF. Keamanan aplikasi web harus dipertimbangkan sejak tahap perancangan dan terus diperbarui seiring berjalannya waktu.