Laporan Uji Pertahanan darunnajah.ac.id: Resource Exhaustion atau Denial of Service (DoS) Akibat Scanning dan Enumeration

Ringkasan Masalah

Selama pengujian untuk menguji pertahanan domain darunnajah.ac.id, saya berhasil **membypass WAF Cloudflare** yang melindungi situs web target. Setelah menggunakan kombinasi alat untuk **scanning**, **enumeration**, dan **SQL injection testing**, server target **menjadi tidak responsif dan down**. Hal ini menunjukkan adanya **kerentanannya** pada server dalam menangani beban permintaan yang tinggi atau eksploitasi otomatis, yang menyebabkan **overload** dan **downtime**.

Langkah Pengujian yang Dilakukan

1. **Bypass WAF Cloudflare**:
    - Menggunakan **ProxyChains** untuk menyembunyikan IP asli dan menghindari deteksi oleh WAF Cloudflare.
    - Melakukan pemindaian dan eksploitasi dengan alat seperti **nmap**, **sqlmap**, dan **burp suite** untuk menganalisis aplikasi web dan mengidentifikasi parameter yang rentan terhadap SQL Injection.
2. **Scanning dan Enumeration**:
    - Menggunakan **nmap** untuk pemindaian port dan layanan yang terhubung.
    - Melakukan **SQL Injection testing** dengan **sqlmap** menggunakan opsi untuk eksploitasi otomatis pada parameter yang teridentifikasi.
    - Melakukan **manual enumeration** pada beberapa endpoint menggunakan **burp suite** dan alat lain untuk menemukan potensi celah.
3. **Hasil yang Terjadi**:
    - Setelah menjalankan alat dengan kecepatan pemindaian yang tinggi, server mulai menunjukkan **penurunan kinerja** dan akhirnya menjadi **down** atau tidak responsif.
    - Mencoba mengakses aplikasi web menunjukkan **timeout error** atau **tidak dapat terhubung** selama beberapa menit setelah pengujian.

Analisis Masalah

- **Overloading Server**: Permintaan yang berasal dari alat otomatis seperti **nmap**, **sqlmap**, dan **burp suite** menghasilkan volume lalu lintas yang sangat tinggi dalam

waktu singkat, yang kemungkinan besar menyebabkan **resource exhaustion** pada server backend.

- **Rate Limiting atau WAF Misconfiguration**: Meskipun WAF Cloudflare berhasil dilewati, server backend sepertinya tidak memiliki mekanisme **rate limiting** atau **load balancing** yang efektif untuk mengatasi jumlah permintaan berulang yang tinggi, yang menyebabkan server menjadi **tidak responsif**.

- **Lack of Resilience**: Server backend mungkin tidak cukup **tahan terhadap serangan otomatis** atau eksploitasi berat, yang menyebabkan **server crash** atau **downtime**.

Reproduksi Langkah

1. Konfigurasi alat dengan **ProxyChains** untuk menghindari deteksi IP oleh WAF.

2. Gunakan **nmap** untuk pemindaian port yang agresif pada aplikasi target.

3. Jalankan **sqlmap** dengan opsi **-u** dan pilih parameter untuk SQL Injection testing (menggunakan **tamper scripts** jika perlu).

4. Gunakan **burp suite** untuk melakukan **manual enumeration** dan eksploitasi lebih lanjut.

5. **Monitor hasilnya** dengan mencoba mengakses situs atau aplikasi web, dan perhatikan apakah server menjadi **down** atau **tidak responsif**.

Potensi Kerentanannya

- **Denial of Service (DoS)** atau **Resource Exhaustion**: Server backend yang tidak memiliki perlindungan terhadap serangan berulang atau eksploitasi otomatis, yang menyebabkan server kehabisan **sumber daya** (memori, CPU, bandwidth) dan menjadi tidak responsif.

- **Misconfiguration dalam WAF atau Rate Limiting**: Cloudflare WAF mungkin tidak cukup ketat dalam membatasi **jumlah permintaan** yang datang dari satu sumber, atau server tidak mengatur **rate limiting** atau **load balancing** dengan benar, memungkinkan serangan untuk mempengaruhi kinerja server.

Rekomendasi

1. **Menerapkan Rate Limiting pada Server**: Pastikan server membatasi jumlah permintaan yang diterima dari satu IP dalam waktu tertentu untuk menghindari **serangan DDoS** atau flooding dari alat otomatis.

2. **Load Balancing yang Lebih Baik**: Implementasikan **load balancing** untuk mendistribusikan permintaan ke banyak server backend, menghindari overloading satu server.

3. **Optimasi dan Caching**: Gunakan **caching** untuk mengurangi beban pada server dan database backend, serta optimalkan query SQL untuk mencegah **resource exhaustion**.

4. **Penerapan Pembatasan pada WAF**: Meninjau kembali konfigurasi **Cloudflare WAF** dan memastikan bahwa **rate limiting** serta **challenge CAPTCHA** diterapkan dengan benar untuk menghindari serangan otomatis.

5. **Penambahan Pemantauan dan Alerting**: Gunakan sistem **pemantauan server** yang dapat memberikan peringatan ketika penggunaan sumber daya mencapai ambang batas tertentu, agar masalah bisa terdeteksi dan diatasi lebih cepat.

Kesimpulan

Server web yang dilindungi oleh **WAF Cloudflare** dapat **menjadi down** jika tidak dilindungi dengan baik dari **serangan otomatis** dan **permainan yang membebani sumber daya** (seperti SQL Injection testing atau scanning yang agresif). Pemecahan masalah ini memerlukan penguatan pengaturan **rate limiting**, **load balancing**, serta **optimasi sumber daya server**. kerentanan ini berfokus pada **resource exhaustion** dan **DoS vulnerability** yang dapat menyebabkan **downtime** aplikasi web.