ANALISIS LOG SERVER UNTUK MENDETEKSI POLA SERANGAN BRUTE FORCE DAN DICTIONARY ATTACK PADA APLIKASI WEB

Abdul Fathir Zain[1], Putri Nabila[2], Rakhmadi Rahman[3]
abdulfathirzain.241031045@mahasiswa.ith.ac.id[1], putrinabila.241031041@mahasiswa.ith.ac.id[2], rakhmadi.rahman@ith.ac.id[3]
Institut Teknologi Bacharuddin Jusuf Habibie

ABSTRAK

Mekanisme autentikasi pada aplikasi web adalah sasaran utama dalam berbagai serangan siber, dengan Brute Force dan Dictionary Attack sebagai ancaman yang paling persisten. Penelitian ini memberikan analisis forensik yangmendalam terhadap log server web (Apache dan Nginx) guna mengidentifikasi karakteristik khusus dari kedua serangan tersebut. Melalui simulasi terkendali dan menggunakan alat penetrasi standar industri seperti Hydra, penelitian ini menganalisis struktur serangan pada lapisan aplikasi (Layer 7). Hasil penelitian menunjukkan bahwa Brute Force dan Dictionary Attack memiliki jejak digital yang dapat diidentifikasi secara statistik melalui analisis varians waktu antar-permintaan (Inter-Arrival Time) dan distribusi ukuran respons. Penelitian ini juga menyarankan suatu kerangka kerja deteksi yang menggunakan ELK Stack dan algoritma Random Forest, yang terbukti lebih efektif dalam mengurangi tingkat positif palsu (false positive) dibandingkan dengan metode tradisional yang berbasis ambang batas.

Kata Kunci: Brute Force, Dictionary Attack, Forensik Log, Elk Stack, Machine Learning, Keamanan Web.

PENDAHULUAN

Dalam ekosistem digital modern, aplikasi web berperan sebagai dasar bagi interaksi ekonomi dan sosial. Namun, sistem autentikasi yang berfungsi sebagai gerbang verifikasi identitas pengguna tetap menjadi titik kegagalan tunggal utama yang paling sering dieksploitasi. Meskipun metode autentikasi multi-faktor (MFA) mulai diadopsi, sebagian besar sistem warisan (legacy) dan UMKM masih mengandalkan kredensial statis. Keadaan ini diperburuk oleh ketidakmampuan administrator sistem untuk membedakan antara perilaku pengguna yang sah dan serangan otomatis yang canggih.[1]

Permasalahan utama yang dihadapi oleh tim operasi keamanan (SecOps) adalah "kelelahan peringatan" (alert fatigue) yang disebabkan oleh tingginya sinyal positif palsu dari sistem deteksi intrusi (IDS) konvensional. IDS yang berbasis tanda tangan (signature) sering tidak mampu mendeteksi serangan yang memanipulasi header HTTP atau yang menggunakan teknik low-and-slow.[2] Oleh sebab itu, dibutuhkan metode analisis log yang bersifat lebih heuristik dan berbasis perilaku (behavioral) untuk meningkatkan akurasi deteksi.

TUJUAN PENELITIAN

Penelitian ini bertujuan untuk:

1. Mengembangkan taksonomi log serangan yang membedakan pola Serangan Brute Force murni dan Dictionary Attack.
2. Mengevaluasi efektivitas parameter statistik, seperti rentang waktu kedatangan paket dan ukuran respons, dalam mengidentifikasi anomali.
3. Menguji implementasi sistem deteksi terpusat dengan menggunakan ELK Stack dan algoritma Machine Learning dalam lingkungan simulasi.

TINJAUAN PUSTAKA

1. Anatomi Serangan Autentikasi

Dua metode serangan utama yang dibahas memiliki karakteristik fundamental yang berbeda, yaitu:

1. Brute Force Attack: Merupakan usaha deterministik untuk menguji seluruh kemungkinan ruang kunci (keyspace). Metode ini menghasilkan volume permintaan yang sangat tinggi dan pola input yang inkremental. Secara komputasi, serangan ini memerlukan biaya tingi namun menjamin keberhasilan jika diberi waktu yang tidak terbatas.[3]
2. Dictionary Attack: Memanfaatkan faktor probabilitas dengan menggunakan daftar kata sandi (wordlist) yang seringdigunakan atau bocoran data sebelumnya. Serangan ini lebih efektif dan sering kali menunjukkan pola lalu lintas yang lebih rendah dibandingkan dengan Brute Force, membuatnya lebih sulit dideteksi oleh firewall berbasis volume sederhana.[4, 5]

2. Forensik Log Server

Log akses (access log) pada server Apache dan Nginx merupakan dokumen penting dalam investigasi insiden. Elemen utama untuk deteksi meliputi:

1. Kode Status (Status Code): Peningkatan kode 401 (Unauthorized) atau 200 (OK - jika aplikasi mengembalikan halaman login kembali) adalah indikator awal.[6]
2. Ukuran Respons (Response Size): Dalam serangan otomatis, ukuran respons dari halaman yang gagal cenderung statis, berbeda dengan pengguna manusia yang memuat aset bersifat dinamis.
3. User-Agent: Alat serangan sering kali meninggalkan tanda User-Agent default (misalnya "Hydra", "Nmap") jika penyerang lupa melakukan spoofing

METODOLOGI

Penelitian ini menerapkan metode eskperimen simulasi (simulation-based experiment) dalam konteks laboratorium virtual yang tertutup.

1. Desain Eksperimen

Infrastruktur pengujian terdiri atas tiga komponen utama:

1. Target Server: Ubuntu Server 22.04 menjalankan Apache HTTP Server (versi 2.4) dan Nginx (versi 1.18). Aplikasi yang menjadi sasaran meliputi DVWA (Damn Vulnerable Web App) dan instalasi WordPress.
2. Attacker Node: Kali Linux Rolling Edition dilengkapi dengan perangkat serangan seperti Hydra, Medusa, dan skrip Python yang dibuat khusus.
3. Monitoring Node: Sebuah server terpisah menjalankan ELK Stack (Elasticsearch, Logstash, Kibana) versi 8.x yang menerima data log secara real-time melalui Filebeat.

2. Skenario Serangan

Tiga skenario dioperasikan untuk menghasilkan dataset log:

2. Skenario A (Brute Force Klasik): Serangan dengan intensitas tinggi menggunakan Hydra tanpa batasan kecepatan (rate limiting).
- Command: hydra -l admin -P rockyou.txt target_ip http-post-form...
3. Skenario B (Dictionary Attack Terarah): Serangan memanfaatkan daftar kata sandi yang terbatas (Top 100) dengan menggunakan skrip Python.
4. Skenario C (Low-and-Slow): Serangan yang melibatkan penyisipan jeda waktu (jitter) secara acak antara 10-60 detik untuk menghindari deteksi berbasis ambang

batas waktu.

3. Pra-pemrosesan Data

Data log mentah diproses melalui Grok filter pada Logstash untuk mengekstraksi fitur:

1. Timestamp (dikonversi menjadi Inter-Arrival Time).
2. Client IP dan User-Agent.
3. HTTP Method dan URL Resource.
4. Response Code dan Response Size (Bytes).

HASIL DAN PEMBAHASAN

1. Analisis Karakteristik Log

Hasil analisis log menunjukkan perbedaan pola yang signifikan antara kedua jenis serangan lalu lintas normal.

A. Pola Brute Force (Skenario A)

Pada log Apache, serangan Hydra menghasilkan pola "gergaji" (sawtooth) pada grafik frekuensi per detik.

1. Volume: Mencapai 10-50 permintaan per detik dari satu IP.
2. Konsistensi: Interval waktu antar-permintaan (Inter-Arrival Time/IAT) memiliki varians mendekati nol ($\sigma^2 \approx 0$).
3. Log Sample:192.168.1.5 - - "POST /dvwa/login.php HTTP/1.1" 200 432 "-" "Mozilla/4.0 (Hydra)"192.168.1.5 - - "POST /dvwa/login.php HTTP/1.1" 200 432 "-" "Mozilla/4.0 (Hydra)"Indikator kritis di sini adalah kode status 200 dengan ukuran 432 bytes yang berulang terus-menerus, menandakan kegagalan login yang konsisten pada aplikasi yang tidak menggunakan kode 401.

B. Pola Dictionary Attack & Low-andSlow (Skenario B dan C)

Serangan ini lebih sulit dideteksi secara volumetrik karena laju permintaannya rendah. Namun, analisis statistik IAT mengungkapkan anomali:

1. Distribusi Uniform: Jika penyerang menggunakan fungsi sleep(random(10,60)), distribusi IAT akan terlihat seragam (uniform).
2. Perbandingan Manusia: Interaksi manusia asli cenderung mengikuti distribusi Pareto atau bursty (berkelompok), bukan seragam atau konstan.

2. Evaluasi Metode Deteksi

A. Deteksi Berbasis Aturan (Fail2Ban)

Pengujian menggunakan konfigurasi standar Fail2Ban (maxretry = 3, findtime = 600) menunjukkan efektivitas tinggi terhadap Skenario A, memblokir IP kurang dari 5 detik. Namun, metode ini gagal total mendeteksi Skenario C (Low-and-Slow) karna frekuensi serangan berada dibawah ambang batas pemicu.

B. Deteksi Berbasis Mechine Learning (Random Forest)

Penelitian ini telah melatih model Random Forest menggunakan fitur statistik (Mean IAT, Variance IAT, Ratio 4xx).

1. Akurasi : 9.33%
2. Recal : 98.3%
3. False Positive : var_iat (Varians IAT) terbukti menjadi alat yang paling efektif untuk membedakan antara bot lambat dan manusia. Bot yang menunjukkan perilaku lambant seringkali tetap memiliki pola waktu yang lebih teratur secara matematis jika dibandingkan dengan manusia.

3. Visualisasi dan Implementasi

Implementasi pada Kibana memungkinkan untuk melakukan pemantauan secara visual. Dashboard "Auth Security" yang dirancang menunjukkan peta panas GeoIP dan pola upaya login yang gagal. Penandaan (tagging) otomatis pada Logstash untuk permintaan POST menuju login.php yang menghasilkan respons statis terbukti efektif dalam menyaring noise log..

KESIMPULAN

Penelitian ini menyimpulkan bahwa meskipun Brute Force dan Dictionary Attack memiliki tujuan yang sama, jejak log yang dihasilkan berbeda secara fundamental. Deteksi berbasis volume (threshold) tidak lagi memadai untuk menangani serangan modern. Transisi menuju analisis perilaku menggunakan statistik Inter-Arrival Time dan ukuran respons adalah kunci untuk meningkatkan akurasi deteksi. Integrasi Machine Learning dalam pipeline analisis log (seperti ELK Stack) menawarkan solusi yang menjanjikan untuk mengurangi positif palsu.

Saran

1. Konfigurasi Log : Administrator disarankan mengaktifkan $request_time dan $upstream_response_time pada Nginx untuk mendeteksi serangan lambat. [18]
2. Mitigasi Bertingkat: Gunakan WAF untuk memblokir pola serangan dikenal, Rate Limiting untuk membatasi kecepatan, dan analisis log perilaku untuk mendeteksi anomali yang lolos.
3. Adopsi MFA : Solusi teknik pamungkas untuk mencegah dampak serangan ini adalah penggunaan Autentikasi Multi-Faktor, yang meniadakan efektivitas penebakan kata sandi. [4]

DAFTAR PUSTAKA

Aji, R. P., et al. (2023). "Monitoring Website Dashboard System Update with Wazuh Technology".
Splunk. (2024). "Brute Force Attacks: Definition and Prevention".
Rublon. (2024). "Brute Force vs. Dictionary Attack: Key Differences".
ResearchGate. (2024). "Payload size histograms in typical HTTP brute force attack".
KJAR. (2024). "Real-time IDS based on Weblog Analysis using Random Forest".
Datascientest. (2024). "Everything about Brute Force Attack and Hydra Tool".