JURNAL SISFOTENIKA
Vol. 16 No. 1, Januari 2026
p-ISSN: 2087-7897 ; e-ISSN : 2460-5344
DOI : 10.30700/sisfotenika.v16i1.604

Implementasi JemagoWeb untuk Perlindungan File Web dari Serangan Siber Judi Online

Implementation of JemagoWeb for Web File Protection Against Online Gambling Cyber Attacks

Febryan Hari Purwanto*[1], Firmansyah Alfarisi[2]
[1,2]Sekolah Tinggi Kesehatan Al-Fatah Bengkulu; Jl. Indragiri gang 3 Serangkai, (0376) 27508
e-mail: *[1]fharipurwanto@gmailcom, [2]firmansyahalfarisi24@gmail.com

Abstrak

Serangan siber yang menargetkan file web melalui aksi upload, pengeditan, dan penghapusan tidak sah sering terjadi pada website, khususnya yang menjadi sasaran hacker judi online. Penelitian ini mengembangkan dan mengimplementasikan JemagoWeb, sebuah sistem proteksi file web berbasis cron job yang mampu melakukan deteksi otomatis setiap satu menit terhadap perubahan file yang mencurigakan. Sistem ini secara proaktif mencegah upload file ilegal dengan mekanisme auto-delete, serta memulihkan file yang hilang atau termodifikasi dengan membandingkan kondisi file saat ini dengan file backup asli. Nama JemagoWeb berasal dari kata "jemago" dalam bahasa Rejang, yang berarti "menjaga", sehingga mencerminkan fungsi sistem sebagai penjaga keamanan dan integritas file web. Pengujian dilakukan pada lingkungan shared hosting yang rentan terhadap serangan file berbahaya, menunjukkan efektivitas JemagoWeb dalam menjaga integritas dan ketersediaan file web secara real-time. Dengan pendekatan otomatis dan berbasis waktu, JemagoWeb memberikan solusi praktis dalam mitigasi serangan siber judi online, meningkatkan keamanan website tanpa mengganggu kinerja hosting.

Kata kunci*—3-5 Serangan siber, judi online, keamanan file web, cronjob, pemulihan file otomatis*

Abstract

Cyberattacks targeting web files through unauthorized uploads, edits, and deletions are common, especially on websites targeted by online gambling hackers. This study develops and implements JemagoWeb, a web file protection system based on a cron job that performs automatic detection every minute of suspicious file changes. The system proactively prevents illegal file uploads via an auto-delete mechanism and restores missing or modified files by comparing current files to original backups. The name JemagoWeb is derived from the Rejang word "jemago," meaning "to guard," reflecting the system's role as a guardian of web file security and integrity. Testing conducted in a shared hosting environment vulnerable to malicious file attacks demonstrated JemagoWeb's effectiveness in maintaining real-time file integrity and availability. By employing an automated, time-based approach, JemagoWeb offers a practical solution to mitigate online gambling cyberattacks, enhancing website security without compromising hosting performance.

Keywords—*3-5 Cyber attack, Online gambling, Web file security, Cronjob, Automatic file recovery*

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mendorong digitalisasi di berbagai sektor, menjadikan situs web sebagai salah satu sarana utama untuk penyampaian informasi dan layanan digital. Namun, seiring meningkatnya ketergantungan terhadap layanan berbasis web, potensi ancaman terhadap keamanan sistem juga semakin kompleks [1]. Salah satu bentuk serangan yang kini marak terjadi adalah serangan siber melalui manipulasi file web, seperti mengunggah file berbahaya, mengedit konten secara ilegal, dan menghapus file penting [2]. Serangan semacam ini sering menargetkan website yang berjalan pada layanan shared hosting, karena sifatnya yang berbagi sumber daya dan memiliki tingkat kontrol keamanan yang terbatas.

Salah satu ancaman nyata yang semakin sering ditemukan adalah aktivitas peretasan yang bertujuan untuk menyisipkan situs judi online ilegal ke dalam server web korban. Teknik yang digunakan oleh pelaku biasanya melibatkan akses ilegal untuk mengunggah file web berisi script judi, menyamarkan file berbahaya agar tampak seperti file asli, hingga menghapus atau mengganti file penting dalam sistem. Jika tidak segera terdeteksi, serangan semacam ini tidak hanya merusak integritas data, tetapi juga mencoreng reputasi website dan berpotensi menimbulkan kerugian hukum.

Fenomena serangan judi online di Indonesia sering kali menyasar domain pemerintah (.go.id) dan pendidikan (.edu) melalui teknik *Web Defacement*. Berdasarkan penelitian Pratama dkk. (2025), penyisipan konten ilegal seperti promosi judi daring dilakukan untuk memanfaatkan reputasi domain guna meningkatkan peringkat SEO situs ilegal tersebut [3]. Dampaknya bukan hanya pada tampilan visual, tetapi juga pada integritas data yang dapat menurunkan kepercayaan publik terhadap instansi terkait [4].

Ancaman terhadap keamanan situs web, terutama penyisipan konten ilegal seperti judi online, terus meningkat seiring dengan tingginya kerentanan pada infrastruktur hosting, di mana Bimandaru, et al. (2023) [5] mengonfirmasi melalui analisis penetrasi black box bahwa layanan shared hosting sering memiliki konfigurasi keamanan yang lemah sehingga mudah dieksploitasi. Upaya mitigasi lebih lanjut untuk serangan spesifik seperti web defacing judi online telah dilakukan oleh Pahlevi, et al. (2025) [2] yang mengusulkan integrasi Wazuh SIEM dan Snort IDS untuk deteksi real-time berbasis signature, serta pendekatan analisis log server menggunakan algoritma Isolation Forest oleh Santoso dan Wahyuni (2024) [6] untuk mendeteksi anomali lalu lintas tanpa pelabelan manual. Di sisi lain, aspek kerahasiaan data pengguna juga diperkuat melalui mekanisme enkripsi sisi klien berbasis ChaCha20-Poly1305 dan Argon2 yang ditawarkan oleh Jehian, et al. (2025) [7]. Meskipun berbagai solusi tersebut terbukti efektif, mayoritas memerlukan akses kontrol penuh (root) terhadap server, instalasi agen yang kompleks, atau sumber daya komputasi tinggi yang umumnya tidak tersedia pada lingkungan shared hosting.

Dalam konteks tersebut, dibutuhkan solusi keamanan yang mampu bekerja secara otomatis dan terus-menerus untuk mendeteksi serta menanggulangi perubahan yang mencurigakan pada file web. Penggunaan pendekatan berbasis cron job menjadi relevan karena mampu menjalankan skrip secara berkala dalam interval waktu tertentu tanpa intervensi manusia. Sistem yang dirancang untuk mendeteksi perubahan file, menghapus file ilegal secara otomatis, dan memulihkan file dari backup asli dinilai efektif dalam menjaga stabilitas dan integritas web.

Penelitian ini mengembangkan JemagoWeb, sebuah sistem proteksi file web berbasis PHP dan cron job yang dapat mendeteksi perubahan file mencurigakan setiap satu menit, melakukan penghapusan otomatis terhadap file yang tidak sah, serta memulihkan file yang hilang atau termodifikasi. Nama "JemagoWeb" diambil dari kata jemago dalam bahasa Rejang yang berarti "menjaga", mencerminkan peran sistem ini sebagai penjaga keamanan file web. Sistem ini

diujicobakan pada lingkungan shared hosting, yang secara umum memiliki tingkat kerentanan lebih tinggi terhadap serangan file berbahaya.

Dengan pendekatan pemantauan otomatis berbasis waktu dan kemampuan pemulihan mandiri, JemagoWeb hadir sebagai solusi praktis dalam mitigasi serangan siber berbasis judi online, tanpa membebani performa hosting. Inovasi ini diharapkan dapat membantu administrator web dalam menjaga integritas dan ketersediaan sistem secara real-time.

2. METODE PENELITIAN

Penelitian ini fokus pada perancangan dan implementasi sistem keamanan berbasis *script* PHP dan cron job. Sistem yang dikembangkan, bernama **JemagoWeb**, dirancang untuk melakukan pemantauan, deteksi, penghapusan otomatis file mencurigakan, serta pemulihan file yang terhapus akibat serangan web, terutama pada server shared hosting.

2.1 Hosting

Hosting adalah layanan penyimpanan yang menampung berbagai jenis data pada website seperti video, gambar, email, kode program, aplikasi, hingga basis data, yang dapat diakses oleh banyak pengguna melalui internet. Ketika seseorang mengunjungi suatu alamat website, sistem akan mengirimkan permintaan ke server hosting. Server tersebut kemudian merespons dengan mengirimkan kembali data dari situs yang diminta dalam bentuk teks, gambar, atau konten lainnya [5].

2.2 Kemanan Website

Seiring dengan meningkatnya jumlah data yang dipertukarkan melalui internet, keamanan website menjadi aspek yang krusial. Setiap organisasi maupun perusahaan dituntut untuk menjaga kerahasiaan, integritas, dan keaslian data sesuai dengan standar keamanan yang berlaku. Kebutuhan ini muncul akibat tingginya ketergantungan masyarakat terhadap penggunaan website, sehingga pengukuran serta peningkatan terhadap keamanan sistem secara menyeluruh harus dilakukan secara berkelanjutan [8].

2.2.1 File Upload Attack

Fitur unggah file pada aplikasi web sering menjadi titik lemah yang dimanfaatkan oleh peretas untuk mengunggah file berisi skrip berbahaya. File dengan ekstensi seperti .php, .txt atau .xml dianggap lebih rawan, karena server cenderung mengeksekusi jenis file tersebut secara otomatis tanpa memerlukan izin tambahan. Setelah dijalankan, peretas dapat melancarkan berbagai serangan siber, seperti menyisipkan *backdoor* pada sistem, merusak aplikasi *web*, menyebarkan *malware*, hingga melakukan serangan *phishing* [9].

2.2.2 Analisis Serangan Web Shell dan Judi Online

Serangan judi online sering kali memanfaatkan kerentanan *File Upload* untuk menanamkan *Web Shell* atau *backdoor* pada server. Penyerang menggunakan *Web Shell* untuk melakukan eksekusi perintah jarak jauh seperti modifikasi file indeks atau pembuatan direktori baru yang berisi konten judi [10]. Mitigasi terhadap serangan ini memerlukan pemantauan integritas file secara berkala. Teknik *File Integrity Monitoring* (FIM) bekerja dengan membandingkan nilai *hash* file saat ini dengan *baseline* yang telah ditentukan untuk mendeteksi perubahan sekecil apa pun secara otomatis [11].

2.3 Automasi Keamanan Menggunakan Cron Job

Implementasi keamanan pada *shared hosting* memiliki tantangan tersendiri karena keterbatasan akses kontrol dan sumber daya. Penggunaan *cron job* sebagai basis penjadwalan eksekusi skrip keamanan terbukti efektif karena ringan dan stabil [12]. *Cron job* adalah tugas

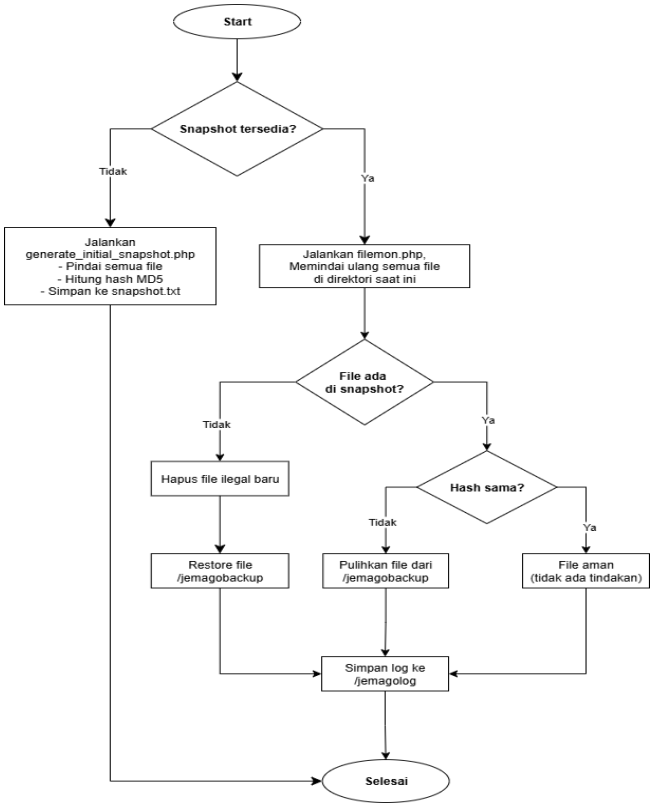
yang dijalankan secara otomatis sesuai jadwal tertentu, yang diatur melalui ekspresi terkode yang mendefinisikan waktu pelaksanaannya. Jadwal ini dapat berupa eksekusi satu kali, berkala, atau dalam selang waktu yang sangat sering. Pengguna dapat mengatur jadwal melalui beberapa bidang seperti menit, jam, bulan, dan lainnya, tergantung pada alat *cron* yang digunakan. Format ini umumnya mengacu pada standar awal dari utilitas *cron* di sistem Unix sejak tahun 1975. Namun, setiap alat *cron* dapat memiliki perbedaan dalam jumlah bidang maupun bentuk ekspresinya. Misalnya, *cron* versi lama tidak menyediakan bidang untuk detik, sementara versi modern sudah mendukungnya. Perbedaan ini muncul karena kebijakan desain yang diambil oleh masing-masing pengembang alat *cron* tersebut [13].

2.4 Hashing MD5

Algoritma MD5 mampu menerima input dengan panjang tak terbatas dan menghasilkan ringkasan pesan (message digest) sepanjang 128 bit. Secara teori, tidak mungkin dua pesan yang berbeda menghasilkan ringkasan yang identik (Rivest, 1992). Algoritma ini dioptimalkan untuk berjalan secara efisien pada arsitektur 32-bit dan tidak membutuhkan tabel substitusi berukuran besar, sehingga prosesnya tetap cepat. Meskipun kinerjanya lebih lambat dibandingkan MD4, MD5 dikembangkan sebagai solusi yang lebih aman karena MD4 telah dianggap rentan terhadap serangan [14].

2.5 Flowchart Sistem JemagoWEB

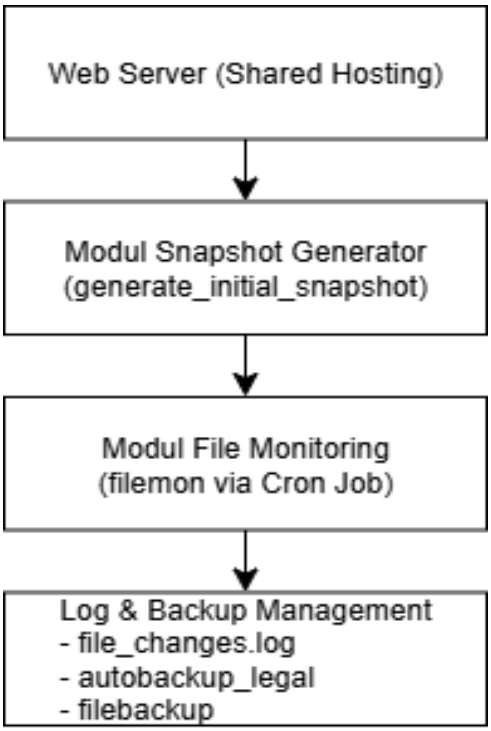
Flowchart merupakan alat visual yang berfungsi untuk memetakan berbagai elemen dalam suatu sistem informasi secara sistematis, ringkas, dan mudah dipahami. Diagram ini menggambarkan urutan proses bisnis serta alur pergerakan dokumen di dalam suatu organisasi. Secara keseluruhan, flowchart bertindak sebagai representasi grafis yang menunjukkan sistem, prosedur, dan mekanisme kontrol internal yang sedang dijalankan oleh perusahaan [15].



Gambar 1. *Flowchart* Sistem JemagoWEB

Flowchart sistem JemagoWEB menggambarkan mekanisme deteksi dan mitigasi file ilegal pada direktori web melalui dua tahapan utama, yaitu proses inisialisasi snapshot dan pemantauan integritas file secara berkala. Pada tahap awal, sistem melakukan pemeriksaan terhadap ketersediaan file snapshot. Apabila snapshot belum terbentuk, modulgenerate_initial_snapshot.php dijalankan untuk memindai seluruh file dalam direktori, menghitung nilai hash MD5, serta menyimpan hasilnya pada berkas snapshot.txt sebagai baseline integritas file. Setelah baseline tersedia, sistem beralih ke tahap pemantauan dengan mengeksekusi modul filemon.php, yang berfungsi untuk melakukan pemindaian ulang dan membandingkan kondisi file terhadap data pada snapshot. Apabila ditemukan file baru yang tidak terdaftar pada snapshot, sistem mengklasifikasikannya sebagai file ilegal, kemudian menghapus serta memulihkannya dengan versi asli dari direktori /jemagobackup. Sementara itu, apabila file terdaftar tetapi nilai hash MD5 tidak sesuai, maka dianggap telah dimodifikasi dan secara otomatis dipulihkan dari cadangan. Sebaliknya, file yang memiliki nilai hash identik dinyatakan aman sehingga tidak memerlukan tindakan korektif. Seluruh aktivitas sistem, baik berupa penghapusan, pemulihan, maupun status aman, secara konsisten dicatat ke dalam direktori /jemagolog untuk keperluan dokumentasi dan analisis lebih lanjut. Dengan demikian, JemagoWEB mampu menjaga integritas direktori web secara real-time sekaligus menyediakan mekanisme mitigasi proaktif terhadap ancaman siber berbasis file.

2.6 Arsitektur Sistem JemagoWEB



Gambar 2. Arsitektur Sistem JemagoWEB

Arsitektur JemagoWEB dirancang untuk memantau dan melindungi file pada direktori publik hosting dari potensi unggahan ilegal. Sistem terdiri dari beberapa komponen utama, yaitu folder yang dipantau (/home/username/public_html), folder backup utama (/home/username/filebackup), folder autobackup legal (/home/username/autobackup_legal), file log aktivitas (file_changes.log), dan snapshot file hash (snapshot.json). Pemantauan dijalankan secara otomatis melalui **cron job** yang memicu skrip filemonpmb.php setiap menit. Pemanfaatan cron job dipilih karena sifatnya yang efisien dalam menjalankan proses terjadwal tanpa

memerlukan intervensi manual, sehingga sistem mampu melakukan deteksi, pencatatan, dan mitigasi secara konsisten.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Implementasi

Sistem JemagoWEB telah berhasil diimplementasikan pada lingkungan shared hosting dengan konfigurasi cron job yang dijalankan setiap satu menit. Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi perubahan file secara real-time, menghapus file ilegal yang disisipkan oleh penyerang, serta memulihkan file yang dimodifikasi tanpa mengganggu ketersediaan layanan web.

Mekanisme awal dimulai dengan pembuatan baseline integritas melalui generate_initial_snapshot.php, yang menghasilkan berkas snapshot berisi daftar file dan nilai hash MD5. Selanjutnya, modul filemonpmb.php berjalan secara otomatis untuk membandingkan kondisi file terkini dengan snapshot yang telah tersimpan.

Implementasi JemagoWEB menunjukkan bahwa sistem dapat dijalankan secara stabil pada lingkungan shared hosting tanpa memerlukan konfigurasi khusus di sisi server. Selama periode pengujian, cron job mampu mengeksekusi script monitoring secara konsisten setiap satu menit tanpa mengalami kegagalan proses atau keterlambatan eksekusi yang signifikan. Hal ini membuktikan bahwa pendekatan berbasis penjadwalan waktu cukup andal untuk diterapkan pada infrastruktur hosting dengan sumber daya terbatas.

Selain itu, penggunaan file snapshot sebagai baseline integritas terbukti mempermudah proses identifikasi perubahan file, karena sistem tidak perlu menyimpan seluruh struktur direktori di memori. Dengan hanya membandingkan nilai hash, proses monitoring dapat dilakukan secara efisien dan tetap akurat dalam mendeteksi manipulasi file yang berukuran kecil maupun besar.

3.2 Logika Sistem JemagoWEB

Logika kerja sistem JemagoWEB dirancang untuk mendeteksi dan menanggapi perubahan file secara otomatis. Proses dimulai dengan pembacaan *snapshot* yang berisi hash file dari kondisi awal. Selanjutnya, sistem menghitung hash file saat ini dan membandingkannya dengan data pada *snapshot*. Jika ditemukan file baru yang mencurigakan atau perubahan pada file yang sudah ada, sistem langsung menandainya sebagai potensi ancaman. Logika utama mencakup:

- Pembuatan snapshot awal untuk mendefinisikan integritas file.
- Perbandingan hash setiap file dengan data pada *snapshot*.
- Identifikasi perubahan, baik penambahan, modifikasi, maupun penghapusan.
- Tindakan otomatis, berupa penghapusan file ilegal dan pemulihan file dari folder *backup*.
- Pencatatan aktivitas ke dalam file log sebagai bentuk dokumentasi.

Logika sistem JemagoWEB dirancang dengan pendekatan deterministik, di mana setiap perubahan file akan selalu menghasilkan keputusan yang sama berdasarkan hasil perbandingan hash dan status keberadaan file. Pendekatan ini mengurangi risiko kesalahan klasifikasi serta memudahkan proses evaluasi ketika terjadi kesalahan sistem atau false positive.

Dengan membagi proses menjadi tahap inisialisasi snapshot dan tahap monitoring berkala, sistem mampu memisahkan kondisi normal dan kondisi anomali secara jelas. Pemisahan ini juga memudahkan administrator dalam melakukan pemeliharaan, karena snapshot dapat diperbarui kapan saja ketika dilakukan perubahan legal pada struktur website.

Potongan *sourcecode* dapat digambarkan sebagai berikut:

```php
$watchDir = '/home/username/public_html'; // Path direktori root yang diawasi
$snapshotFile = __DIR__ . '/snapshot.json';

// Fungsi untuk mendapatkan snapshot direktori
function getDirectorySnapshot($dir) {
    $files = [];
    $rii = new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir));

    foreach ($rii as $file) {
        if ($file->isFile()) {
            $files[$file->getPathname()] = md5_file($file->getPathname());
        }
    }

    return $files;
}

// Generate snapshot
echo "Create Initial snapshot file from : $watchDir\n";
$snapshot = getDirectorySnapshot($watchDir);
```

Gambar 3. *Generate Initial Snapshot*

Gambar 3 memperlihatkan bahwa logika ini bertugas memindai seluruh file di direktori public_html, menghitung hash MD5 setiap file, lalu menyimpannya ke snapshot.json sebagai *baseline* integritas. Proses pembuatan snapshot dilakukan secara otomatis tanpa interaksi pengguna, sehingga meminimalkan risiko kesalahan konfigurasi manual. Seluruh file diproses secara berurutan, dan hasil perhitungan hash disimpan dalam format terstruktur yang mudah dibaca kembali oleh modul monitoring.

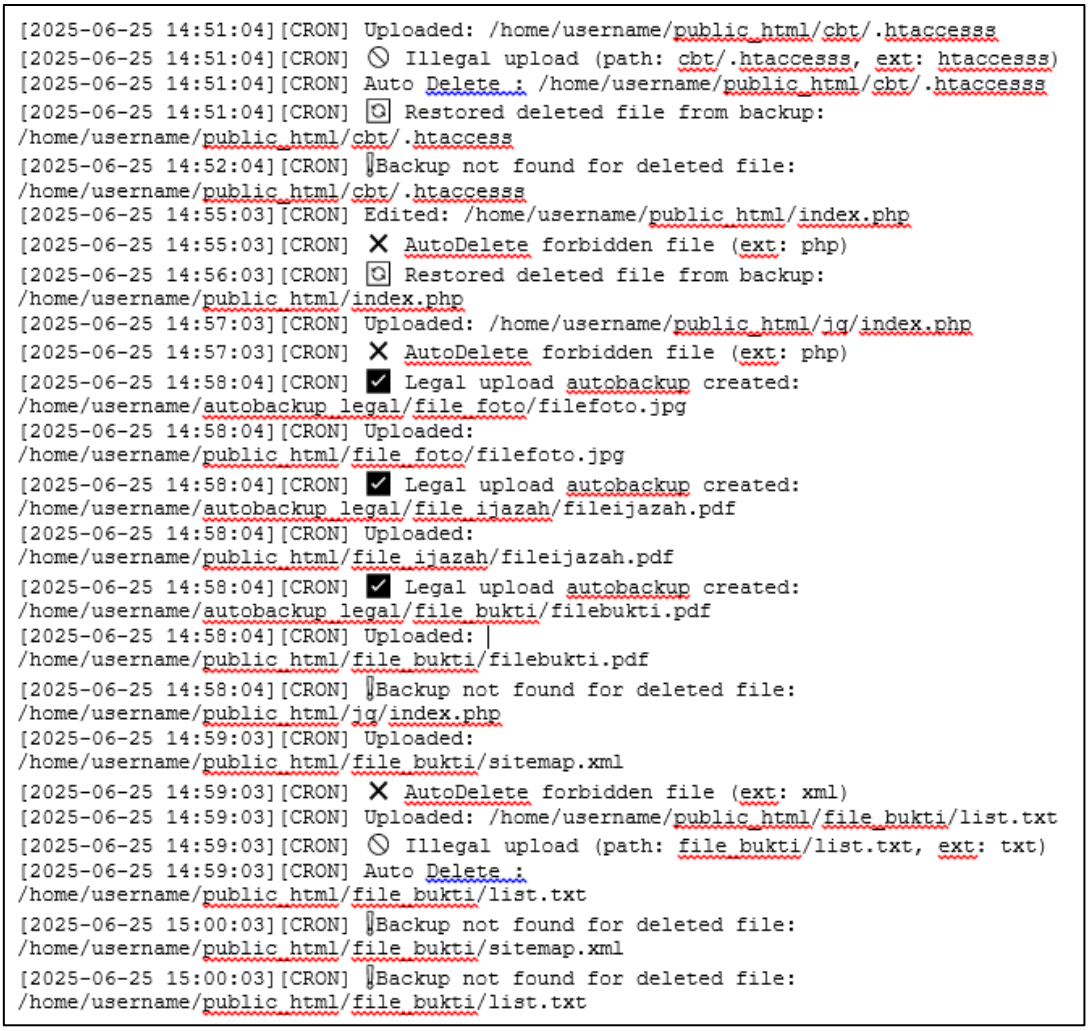
```php
// Ambil snapshot terbaru saat ini dari direktori yang dipantau
$currentSnapshot = getDirectorySnapshot($watchDir);

// Bandingkan snapshot lama dan baru untuk mendeteksi perubahan file
$changes = [];
foreach ($currentSnapshot as $file => $hash) {
    if (!isset($prevSnapshot[$file])) {
        $changes[] = ['type' => 'Uploaded', 'file' => $file]; // File baru ditambahkan
    } elseif ($prevSnapshot[$file] !== $hash) {
        $changes[] = ['type' => 'Edited', 'file' => $file];   // File diubah
    }
}
foreach ($prevSnapshot as $file => $hash) {
    if (!isset($currentSnapshot[$file])) {
        $changes[] = ['type' => 'Deleted', 'file' => $file];  // File dihapus
    }
}
```

Gambar 4. *File monitoring*

Pada Gambar 4 terlihat bahwa proses monitoring tidak hanya berfokus pada keberadaan file, tetapi juga pada integritas isi file. Walaupun nama file tidak berubah, perbedaan kecil pada konten akan menghasilkan nilai hash yang berbeda dan langsung terdeteksi oleh sistem. Hal ini penting karena serangan judi online sering kali dilakukan dengan memodifikasi file indeks yang sudah ada agar tidak mudah dicurigai.

3.3 Pengujian Log Aktivitas JemagoWEB



Gambar 5. *File Log Aktivitas*

Gambar 5 menampilkan hasil pengujian JemagoWEB yang direpresentasikan melalui log aktivitas sistem selama proses simulasi serangan dan unggahan file pada direktori public_html. Log ini menjadi bukti bahwa sistem berjalan sesuai dengan rancangan, khususnya dalam mendeteksi perubahan file, mengklasifikasikan ancaman, serta melakukan tindakan otomatis berupa penghapusan, pemulihan, dan pencadangan.

Berdasarkan catatan log, sistem berhasil mendeteksi file ilegal dengan ekstensi terlarang seperti .php, .txt, dan .xml yang kemudian langsung dihapus oleh sistem (*auto-delete*). Selain itu, ketika ditemukan file legal yang mengalami modifikasi, seperti index.php, sistem secara otomatis menghapus versi yang telah berubah dan memulihkan file tersebut dari direktori backup. Sebaliknya, file legal seperti .jpg dan .pdf tidak hanya diizinkan, tetapi juga disalin ke direktori *autobackup_legal* sebagai langkah pengamanan tambahan.

Adanya pesan *Backup not found* menunjukkan kondisi ketika file yang dihapus belum memiliki cadangan, sehingga sistem hanya dapat mencatat kejadian tanpa melakukan pemulihan. Secara keseluruhan, hasil pengujian ini menunjukkan bahwa JemagoWEB mampu melakukan pemantauan dan mitigasi file secara otomatis, konsisten, dan responsif terhadap berbagai pola serangan berbasis unggahan file pada lingkungan shared hosting.

Keberadaan log aktivitas juga memudahkan proses penelusuran kronologi serangan, mulai dari waktu unggahan file ilegal hingga tindakan penghapusan atau pemulihan yang dilakukan sistem. Informasi ini dapat digunakan sebagai dasar evaluasi kebijakan keamanan

hosting maupun sebagai bahan analisis untuk meningkatkan konfigurasi sistem di masa mendatang.

Selain itu, pencatatan yang konsisten memungkinkan administrator untuk membedakan antara perubahan file yang sah (misalnya akibat pembaruan website) dan perubahan yang berasal dari aktivitas mencurigakan, sehingga mengurangi risiko kesalahan interpretasi terhadap kondisi sistem.

3.4 Evaluasi Kinerja Sistem

Hasil pengujian menunjukkan bahwa JemagoWEB mampu mendeteksi perubahan file dan aktivitas mencurigakan dalam waktu kurang dari satu menit setelah pemicu cron job berjalan. Respons cepat ini menjadikan sistem efektif dalam menekan potensi kerusakan website akibat penyisipan file ilegal, seperti skrip judi online dan malware berbasis upload. Dibandingkan pemeriksaan manual, mekanisme otomatis ini lebih efisien karena bekerja secara konsisten tanpa ketergantungan pada operator, sehingga mampu menjaga integritas file secara berkelanjutan.

Meskipun demikian, evaluasi juga mengidentifikasi beberapa keterbatasan. JemagoWEB hanya berfokus pada pemantauan di tingkat file sehingga tidak dapat menangani serangan yang berorientasi pada database, manipulasi server configuration, ataupun eksploitasi di luar direktori yang dipantau. Selain itu, jika snapshot awal dibentuk dalam kondisi server yang sudah terinfeksi, sistem berpotensi menganggap file berbahaya sebagai file yang sah. Dengan demikian, pembuatan snapshot harus dilakukan pada kondisi server yang benar-benar bersih.

Catatan penting lainnya adalah bahwa efektivitas JemagoWEB sangat dipengaruhi oleh tingkat keamanan hosting. Sistem tidak dapat bekerja optimal apabila penyerang telah memperoleh akses ke direktori di luar public_html atau mampu mengeksploitasi konfigurasi panel hosting, termasuk pengaturan cron job. Dalam kondisi tersebut, penyerang dapat memodifikasi lingkungan eksekusi atau bahkan menonaktifkan script monitoring. Oleh karena itu, keberhasilan implementasi JemagoWEB harus didukung oleh infrastruktur hosting yang memiliki standar keamanan tinggi, baik pada level file system maupun konfigurasi server.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

1. JemagoWeb berhasil dikembangkan sebagai system keamanan berbasis PHP dan *cron job* yang mampu melakukan pemantauan file web secara otomatis pada lingkungan *shared hosting*.
2. Sistem JemagoWeb mampu mendeteksi perubahan file dalam interval satu menit, mengidentifikasi unggahan file ilegal, serta melakukan tindakan mitigasi berupa penghapusan otomatis dan pemulihan file dari cadangan.
3. Hasil pengujian menunjukkan bahwa mekanisme deteksi berbasis *hash* MD5 dan *snapshot* file bekerja secara konsisten dalam menjaga integritas file web serta mengurangi risiko penyisipan skrip berbahaya, termasuk konten judi online.
4. Pencatatan aktivitas melalui sistem log memberikan dokumentasi yang jelas terhadap setiap kejadian perubahan file, sehingga mempermudah proses audit keamanan dan analisis insiden.
5. Meskipun efektif, JemagoWeb memiliki keterbatasan karena hanya beroperasi pada level file dan belum mencakup deteksi serangan berbasis database maupun eksploitasi konfigurasi server.

6. Keberhasilan implementasi JemagoWeb sangat bergantung pada tingkat keamanan infrastruktur hosting, sehingga sistem akan bekerja lebih optimal apabila didukung oleh konfigurasi hosting yang aman.

5. SARAN

Untuk menutup kekurangan penelinitan ini, disarankan menambahkan sistem untuk memindai seluruh direktori yang ada pada hosting baik pada public html atau direktori di atasnya untuk memastikan bahwa hosting bersih dari malware, backdoor dan file-file berbahaya lainnya sebelum dilakukan implementasi jemagoweb. Untuk itu kami juga akan melakukan penelitian lebih lanjut untuk pembuatan aplikasi pemindai file-file berbahaya pada hosting termasuk file yang terindikasi judi online yang akan diberi nama JemagoScan.

DAFTAR PUSTAKA

[1] N. H. Sinaga, D. Irmayani, and M. N. S. Hasibuan, "Mengoptimalkan Keamanan Jaringan: Memanfaatkan Kecerdasan Buatan Untuk Meningkatkan Deteksi Dan Respon Ancaman," *J. Ilmu Komput. dan Sist. Inf. (JIKOMSI*, vol. 7, no. 2, pp. 364–369, 2024.

[2] M. Rizky, R. Pahlevi, C. Umam, and L. B. Handoko, "Deteksi dan Pencegahan Web Defacing Judi Online dengan Wazuh SIEM dan Snort IDS Berbasis Signature," *J. Algoritm.*, vol. 22, no. 1, pp. 197–208, 2025, doi: 10.33364/algoritma/v.22-1.2220.

[3] A. M. Pratama, M. Data, and W. Yahya, "Deteksi Konten Ilegal Pada Situs Web Menggunakan Elasticsearch," *J. Pengemb. Teknol. ...*, vol. 9, no. 10, pp. 1–9, 2025.

[4] Y. Raharja, "JIP (Jurnal Informatika Polinema) Implementasi Metode Osint Untuk Mengidentifikasi Serangan Judi Online Pada Website," *JIP (Jurnal Inform. Polinema)*, vol. 10, no. 3, pp. 359–364, 2024.

[5] A. Bimandaru, A. Alamsyah, and A. Nugroho, "ANALISIS PENGUJIAN PENETRASI PADA LAYANAN HOSTING MENGGUNAKAN METODE BLACK BOX (Studi kasus : Blogspot, Wordpress dan Shared Hosting)," *Foristek*, vol. 14, no. 1, 2023, doi: 10.54757/fs.v14i1.238.

[6] D. B. Santoso *et al.*, "SESTEM LOG WEB SERVER SEBAGAI PENDETEKSI ANOMALI MENGGUNAKAN ISOLATION FOREST WEB SERVER LOG SYSTEM AS AN ANOMALY DETECTOR USING," vol. 4, no. 3, pp. 90–96, 2024.

[7] N. T. Jehian *et al.*, "PENGEMBANGAN SISTEM KEAMANAN DATA BERBASIS WEB MENGGUNAKAN KOMBINASI ALGORITMA CHACHA20-POLY1305 DAN ARGON2," vol. 13, no. 3, 1958.

[8] P. Mitra Purba, Azrah Cipta Amandha, Riyan Hidayah Purnama, and Ali Ikhwan, "Analisis Keamanan Website Prodi Sistem Informasi Uinsu Menggunakan Metode Application Scanning," *J. Inform. Teknol. dan Sains*, vol. 4, no. 4, pp. 325–329, 2022, doi: 10.51401/jinteks.v4i4.2065.

[9] J. T. Informatika and P. N. Indramayu, "Pengujian Keamanan Fitur Upload File Pada Sistem Aplikasi Web," vol. 7, no. 1, pp. 37–42, 2022.

[10] E. S. Alim, N. Nuroji, M. A. Rizkiawan, T. Anhari, and B. Sobari, "Monitoring dan Pencegahan Serangan Judi Online (Slot Gacor) pada Website," *Edumatic J. Pendidik. Inform.*, vol. 8, no. 1, pp. 75–83, 2024, doi: 10.29408/edumatic.v8i1.25267.

[11] A. Herrero and P. De Albeniz, "File integrity monitoring on Linux systems," 2021.

[12] S. Parulian, D. A. Pratiwi, and M. Cahya Yustina, "Ancaman dan Solusi Serangan Siber di Indonesia," *Telecommun. Networks, Electron. Comput. Technol.*, vol. 1, no. 2, pp. 85–92, 2021, [Online]. Available: http://ejournal.upi.edu/index.php/TELNECT/

[13] A. Cimino, "Master ' s Degree programme Computer Science and Information Technology - CM90 CronFrame : A Macro Annotation Cron Job Framework with Web

Server and CLI Tool written in Rust," 2024.
[14] N. Ardian Yulianto *et al.*, "Analisis Kinerja Algoritma MD5, SHA-256, dan Base62 dalam Sistem Pemendekan URL Info Artikel," *J. Manaj. Inform. Sist. Inf. dan Teknol. Komput.*, vol. 3, no. 2, pp. 270–276, 2024, doi: 10.70247/jumistik.vi2.113.
[15] Z. Tuasamu *et al.*, "Analisis Sistem Informasi Akuntansi Siklus Pendapatan Menggunakan DFD Dan Flowchart Pada Bisnis Porobico," *J. Bisnis dan Manajemen(JURBISMAN)*, vol. 1, no. 2, pp. 495–510, 2023.