Analisis Komparatif Kemampuan GPT dan Burp Suite dalam Pengujian Kerentanan Aplikasi Web

Maulina Nur Laila[1], Qoyyimil Jamilah[2], Sintiarani Febyan Putri[3]
Departemen Sistem Informasi, Fakultas Teknologi Elektro Dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember
Email: maulina.2034@gmail.com[1], qoyyimillj@gmail.com[2], sintiap288@gmail.com[3]

ABSTRAK

Keamanan aplikasi web merupakan aspek krusial dalam era digital yang semakin terintegrasi, mengingat semakin kompleksnya ancaman terhadap sistem informasi. Penelitian ini bertujuan untuk melakukan analisis komparatif antara dua pendekatan dalam pengujian kerentanan aplikasi web, yaitu pendekatan praktis menggunakan Burp Suite dan pendekatan teoritis berbasis *Large Language Models* (LLM), khususnya *Generative Pre-trained Transformer* (GPT). Penelitian ini menggunakan metode studi literatur terhadap berbagai jurnal ilmiah, prosiding konferensi, dan dokumentasi teknis yang relevan. Hasil telaah menunjukkan bahwa Burp Suite memiliki keunggulan pada pengujian penetrasi dinamis, eksploitasi manual, serta dukungan modul yang kaya seperti *scanner*, *intruder*, dan *repeater*. Burp Suite juga mendeteksi lebih banyak kerentanan kritis dibandingkan alat lain seperti OWASP ZAP, meskipun masih terbatas dalam deteksi tipe tertentu seperti XSS dan membutuhkan intervensi manusia dalam verifikasi hasil.

Sementara itu, model GPT menunjukkan performa tinggi dalam deteksi kerentanan kode secara statis, dengan studi yang melaporkan F1-score di atas 0,90 pada GPT-4o dan Claude-3.5 Sonnet. GPT juga menunjukkan kemampuan dalam menghasilkan rekomendasi perbaikan, membangun unit test, serta mengotomatisasi tugas-tugas teknis seperti analisis *shell* dan perintah sistem. Namun, model ini belum sepenuhnya andal untuk pengujian penetrasi otomatis karena keterbatasan seperti *false positives*, halusinasi output, dan kehilangan konteks jangka panjang. Analisis perbandingan menyimpulkan bahwa kedua alat memiliki keunggulan dan kekurangan masing-masing yang saling melengkapi. Oleh karena itu, penelitian ini merekomendasikan pendekatan hibrida, di mana GPT dapat digunakan untuk menghasilkan analisis awal dan skenario uji, sedangkan Burp Suite bertindak sebagai alat validasi praktis yang menjalankan pengujian secara langsung dan terstruktur.

Kata Kunci : Burp Suite, Deteksi Kerentanan, GPT, Keamanan Aplikasi Web, *Large Language Models*.

Article History
Received: Juni 2025
Reviewed: Juni 2025
Published: Juni 2025

Plagiarism Checker No 235
Prefix DOI : 10.8734/Kohesi.v1i2.365
Copyright : Author
Publish by : Kohesi

PENDAHULUAN

Dalam era digital yang semakin kompleks, keamanan aplikasi web menjadi isu yang sangat krusial. Seiring meningkatnya ketergantungan pada layanan berbasis web, risiko

terhadap serangan siber seperti *SQL Injection*, *Cross-Site Scripting* (XSS), dan serangan rekayasa sosial turut meningkat. Untuk menghadapi ancaman tersebut, berbagai alat pengujian keamanan telah dikembangkan, salah satunya adalah Burp Suite, yaitu alat uji penetrasi (*penetration testing*) yang digunakan untuk mengidentifikasi serta mengeksploitasi kerentanan pada aplikasi web secara langsung. Burp Suite mendukung beragam fungsi seperti *proxy* analisis, *scanning* otomatis, dan simulasi serangan nyata terhadap sistem target, menjadikannya salah satu alat praktis paling handal di bidang pengujian keamanan aplikasi (Choudhary et al., 2023).

Di sisi lain, perkembangan teknologi kecerdasan buatan mendorong pemanfaatan *Large Language Models* (LLM) seperti GPT dalam ranah keamanan siber. GPT mampu menganalisis kode sumber, mengenali pola berbahaya, dan memberikan saran perbaikan berdasarkan konteks serta data pelatihan yang luas. GPT-4o mampu mendeteksi kelemahan kode dengan nilai F1 mencapai 0,9072 melalui teknik *prompting* bertahap (Bae et al., 2024). Selain itu, studi oleh Khare et al. (2023) mencatat bahwa GPT-3.5 dapat mendeteksi berbagai jenis kerentanan dalam kode sumber dengan rata-rata akurasi sebesar 62,8% dan F1-score mencapai 0,71 (Khare et al., 2023).

Walaupun menjanjikan, penerapan GPT dalam deteksi kerentanan masih menghadapi sejumlah tantangan seperti keterbatasan pemahaman konteks, potensi munculnya *false positives*, dan perlunya penyesuaian lanjutan agar lebih relevan untuk konteks teknis tertentu. Sebaliknya, Burp Suite menawarkan efektivitas dalam uji langsung namun tidak memiliki kemampuan untuk memberikan rekomendasi berbasis pola atau konteks kode yang kompleks seperti GPT. Oleh karena itu, diperlukan kajian komparatif antara GPT dan Burp Suite untuk mengevaluasi efektivitas, keunggulan, dan batasan dari masing-masing pendekatan dalam pengujian kerentanan aplikasi web (Bsharat et al., 2023)

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk mengkaji dan membandingkan dua pendekatan dalam pengujian kerentanan aplikasi web, yakni pendekatan teoritis berbasis *Generative AI* (GPT) dan pendekatan praktis melalui penggunaan Burp Suite. Permasalahan utama yang diangkat adalah sejauh mana efektivitas dan kontribusi masing-masing alat dalam proses deteksi serta pengujian kerentanan, serta bagaimana keduanya dapat saling melengkapi untuk memperkuat sistem keamanan aplikasi web secara menyeluruh. Hasil dari penelitian ini diharapkan dapat memberikan wawasan kepada pengembang dan praktisi keamanan mengenai pemanfaatan teknologi AI dalam mendukung alat keamanan konvensional.

METODE PENELITIAN

Penelitian ini dilaksanakan dengan menggunakan pendekatan studi literatur (*literature study*), yang berfokus pada pengumpulan dan analisis berbagai sumber tertulis yang relevan untuk memahami dan membandingkan dua pendekatan dalam pengujian kerentanan aplikasi web, yaitu pendekatan berbasis *Large Language Models* (LLM) seperti GPT dan pendekatan praktis menggunakan alat pengujian keamanan aplikasi web seperti Burp Suite. Studi literatur ini bertujuan untuk membangun pemahaman konseptual yang kuat berdasarkan temuan-temuan sebelumnya, serta meninjau sejauh mana efektivitas, keunggulan, dan tantangan dari masing-masing pendekatan.

Jenis dan Sumber Data

Penelitian ini menggunakan data sekunder yang diperoleh melalui metode studi literatur. Data yang dikumpulkan berupa informasi, temuan, dan analisis dari berbagai sumber tertulis yang relevan dengan topik penelitian, terutama yang membahas pengujian kerentanan aplikasi web menggunakan pendekatan berbasis kecerdasan buatan (seperti GPT) dan alat konvensional (seperti Burp Suite). Sumber data meliputi jurnal ilmiah, prosiding konferensi, artikel ulasan, serta dokumen teknis yang diterbitkan oleh institusi atau komunitas akademik yang kredibel. Kriteria pemilihan sumber mencakup:

- Relevansi terhadap topik penelitian, khususnya dalam konteks keamanan siber, *vulnerability assessment*, penggunaan AI dalam keamanan aplikasi web, dan pengujian penetrasi sistem
- Kredibilitas sumber, yang ditandai dengan publikasi melalui jurnal *peer-reviewed* atau konferensi ilmiah terakreditasi
- Kemutakhiran, dengan preferensi terhadap publikasi dalam lima tahun terakhir untuk mencerminkan kondisi dan perkembangan teknologi terkini.

Sumber-sumber literatur dapat berasal dari basis data akademik seperti Google Scholar, IEEE Xplore, SpringerLink, ScienceDirect, MDPI, arXiv, dan lainnya. Informasi yang diperoleh dari literatur ini akan dianalisis secara kualitatif dan digunakan sebagai dasar dalam melakukan perbandingan terhadap efektivitas dan karakteristik masing-masing pendekatan pengujian kerentanan.

Teknik Pengumpulan Data

Proses pengumpulan data dilakukan dengan metode penelusuran terstruktur menggunakan kata kunci tertentu pada mesin pencarian akademik seperti Google Scholar dan database jurnal resmi. Beberapa kata kunci yang digunakan antara lain:

- “GPT for vulnerability detection”
- “Burp Suite web security testing”

- “Large Language Models for cybersecurity”
- “Web application penetration testing tools”
- “Comparison of AI and conventional security tools”

Hasil pencarian disaring berdasarkan judul, abstrak, tahun terbit, serta relevansi terhadap fokus penelitian. Artikel yang dipilih harus memuat informasi mengenai metode pengujian keamanan, hasil evaluasi terhadap alat, atau studi penerapan LLM dalam mendeteksi kerentanan.

Teknik Analisis Data

Data yang telah dikumpulkan dianalisis menggunakan pendekatan analisis deskriptif dan komparatif. Tahapan analisis dilakukan sebagai berikut:

- Klasifikasi Sumber: Sumber literatur diklasifikasikan berdasarkan jenis alat yang dibahas (GPT atau Burp Suite), jenis pendekatan yang digunakan (analisis statis, dinamis, atau berbasis AI), serta indikator kinerja yang dinilai (akurasi, efisiensi, fleksibilitas, kemudahan integrasi, dan coverage kerentanan).
- Sintesis Informasi: Setiap artikel yang relevan dibaca secara menyeluruh untuk diidentifikasi poin-poin penting terkait kelebihan, keterbatasan, dan konteks penerapan masing-masing alat.
- Perbandingan Konseptual: Hasil dari literatur kemudian dibandingkan untuk melihat kesamaan dan perbedaan antara pendekatan GPT dan Burp Suite dalam mendeteksi serta menguji kerentanan aplikasi web.
- Penyusunan Tabel Komparatif: Untuk mendukung analisis, dibuat tabel perbandingan yang merangkum fitur-fitur utama dan performa masing-masing alat berdasarkan referensi yang digunakan.

Pendekatan ini memungkinkan peneliti untuk menyajikan hasil analisis secara sistematis dan berbasis bukti, serta menarik kesimpulan yang valid mengenai posisi dan kontribusi masing-masing alat dalam konteks keamanan aplikasi web.

HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dan pembahasan komparatif mengenai kemampuan model GPT dan Burp Suite dalam pengujian kerentanan aplikasi web, berdasarkan studi literatur yang telah dilakukan. Pembahasan akan mencakup fitur, keunggulan, keterbatasan, serta potensi sinergi antara kedua pendekatan ini.

Kemampuan Burp Suite dalam Pengujian Kerentanan Aplikasi Web

Burp Suite adalah alat pengujian penetrasi dan deteksi kerentanan yang dikembangkan oleh Portswigger, dirancang sebagai *all-in-one toolkit* yang komprehensif untuk pengujian keamanan aplikasi web (Mathew dan Benjamin, 2021; Choudhary et al., 2023). Alat ini juga telah berkembang untuk mendeteksi *bug* pada API dan aplikasi seluler (Mathew dan Benjamin, 2021). Burp Suite tersedia dalam tiga edisi: *Community Edition* (gratis dengan fitur terbatas), *Professional Edition* (untuk praktisi individu dengan pengujian manual dan otomatis), dan *Enterprise Edition* (untuk organisasi yang mengintegrasikan pemindaian keamanan ke dalam *software pipelines*) (Mathew dan Benjamin, 2021).

Tabel 1. Modul - Modul Utama dalam Burp Suite (Mathew dan Benjamin, 2021; Choudhary et al., 2023)

Modul	Deskripsi
Target	Menyediakan informasi terperinci tentang aplikasi target dan mengontrol proses pengujian.
Proxy	Berfungsi sebagai *man-in-the-middle web proxy* yang mencegat lalu lintas antara browser dan aplikasi web target, memungkinkan pemantauan dan analisis lalu lintas web.
Scanner	Pemindai kerentanan web canggih yang merayapi konten dan mengauditnya untuk berbagai kerentanan, tersedia di versi profesional.
Intruder	Alat serbaguna untuk mengotomatiskan dan menyesuaikan serangan aplikasi web seperti *brute-force*, *fuzzing*, dan manipulasi parameter.
Repeater	Memungkinkan modifikasi manual dan pengiriman ulang permintaan web, mendukung penanganan sesi untuk pengujian fungsionalitas yang diautentikasi.
Sequencer	Ideal untuk memverifikasi cookies dan elemen sesi lainnya.
Decoder	Untuk mendekode dan mengkodekan data aplikasi.
Comparer	Membandingkan dua bagian data secara visual.
Extender	Memungkinkan pemuatan plugin Burp (BApps) untuk memperluas fitur

Dalam perbandingan kinerja, Burp Suite menunjukkan kekuatan dan kelemahan relatif terhadap alat lain. Sebuah studi menunjukkan bahwa Burp Suite mendeteksi 203 kerentanan dan menghasilkan 203 URL *alerts*, sementara OWASP ZAP mendeteksi 22 kerentanan tetapi menghasilkan 792 URL *alerts*. Burp Suite juga mendeteksi proporsi kerentanan berisiko tinggi

yang lebih tinggi (5 dari 203) dibandingkan OWASP ZAP (1 dari 22). Namun, OWASP ZAP memiliki proporsi kerentanan dengan kepercayaan sedang yang lebih tinggi. Dalam hal waktu pemindaian, Burp Suite membutuhkan sekitar 4 jam dengan rata-rata 10-13 *crashes*, sedangkan OWASP ZAP dapat menyelesaikan dalam 2 jam dengan 6-8 *crashes*. Klasifikasi kerentanan ke dalam OWASP Top 10 juga berbeda; Burp Suite mengidentifikasi 6 kategori, sementara OWASP ZAP 5 kategori. Misalnya, Burp Suite mengidentifikasi 11 kerentanan terkait "*Cryptographic Failures*" yang tidak teridentifikasi oleh OWASP ZAP (Jarupunphol et al., 2023).

Dibandingkan dengan Acunetix dan Netsparker, Burp Suite memiliki kemampuan deteksi *Cross-Site Scripting* (XSS) yang "sangat lemah", namun lebih unggul dalam deteksi kerentanan *security misconfiguration*. Hal ini menunjukkan bahwa tidak ada satu alat pun yang unggul dalam semua jenis kerentanan, sehingga pendekatan multi-alat seringkali diperlukan untuk pengujian keamanan yang komprehensif (Jarupunphol et al., 2023). Burp Suite memberikan nilai paling besar bagi konsultan keamanan independen karena kemudahan penggunaan, fleksibilitas lisensi, dan cakupan fitur yang luas (Mathew dan Benjamin, 2021).

Kemampuan Model Bahasa Besar (GPT) dalam Pengujian Kerentanan Aplikasi Web

Model Bahasa Besar (LLM), khususnya model *Generative Pre-Trained Transformer* (GPT), telah menunjukkan kemampuan mendalam dalam pemahaman teks seperti manusia dan kemampuan yang muncul (*emergent abilities*) seperti penalaran, peringkasan, dan pemecahan masalah domain-spesifik. LLM dilengkapi dengan pengetahuan umum yang luas dan kapasitas untuk penalaran dasar, mampu memahami, menyimpulkan, dan menghasilkan teks yang menyerupai komunikasi manusia (Deng et al., 2024). ChatGPT, sebagai teknologi NLP inovatif, merevolusi interaksi manusia-komputer dengan memanfaatkan algoritma *Machine Learning* (ML) dan *Deep Learning* (DL) (Hadi et al., 2023). LLM adalah alat yang kuat yang membantu tim keamanan siber mengotomatiskan tugas-tugas berulang, mempercepat deteksi dan respons ancaman, serta meningkatkan akurasi tindakan (Kaur et al., 2023).

Aplikasi Model GPT dalam Deteksi Kerentanan Kode Statis

LLM telah banyak digunakan dalam analisis kerentanan, termasuk deteksi kerentanan. GPT-3.5 dapat menandingi, sementara GPT-4 mengungguli metode deteksi *state-of-the-art*. Model GPT dapat digunakan untuk mengidentifikasi kerentanan dan kelemahan perangkat lunak dalam kode sumber (Pelofske et al., 2024). Sebuah studi menunjukkan bahwa GPT-4o dan Claude-3.5 Sonnet secara signifikan mengungguli GPT-3.5 Turbo dalam deteksi kerentanan (Bae et al., 2024).

Meskipun menjanjikan, LLM saat ini tidak cocok untuk pemindaian kerentanan yang sepenuhnya otomatis karena tingkat *false positive* dan *false negative* yang terlalu tinggi. Model

GPT cenderung memiliki bias dalam *false positive*, seringkali salah mengidentifikasi kerentanan tertentu. Misalnya, Mistral-7B-Instruct-v0.1 menunjukkan bias ke CWE-124, Llama-2-70b-chat-hf ke CWE-123, dan Turdus ke CWE-476 (Pelofske et al., 2024).

Kemampuan Model GPT dalam Pengujian Penetrasi Otomatis

Pengujian penetrasi secara tradisional sulit diotomatisasi karena keahlian yang luas yang dibutuhkan oleh profesional manusia. *Framework* pengujian penetrasi otomatis berbasis LLM, seperti PENTEST GPT, memanfaatkan pengetahuan domain yang melimpah untuk mengatasi tantangan ini (Deng et al., 2024).

Keunggulan LLM dalam Sub-tugas (Deng et al., 2024):

a. Penggunaan Alat dan Interpretasi Output: LLM menunjukkan kemahiran dalam menggunakan alat pengujian penetrasi dan menginterpretasikan output-nya, seperti mengonfigurasi nmap dan memahami hasil pemindaian.
b. Identifikasi Kerentanan: LLM menunjukkan pemahaman mendalam tentang jenis kerentanan umum dan menghubungkannya dengan layanan pada sistem target.
c. Analisis dan Generasi Kode: LLM efektif dalam analisis dan generasi kode, terutama dalam tugas-tugas seperti Analisis Kode dan Konstruksi *Shell*. GPT-4, khususnya, unggul dalam interpretasi dan generasi kode.
d. Pembuatan Perintah dan Deskripsi GUI: LLM dapat membuat perintah pengujian yang sesuai dan menjelaskan operasi antarmuka pengguna grafis (GUI) dengan akurat.
e. Prosedur Pengujian Inovatif: Memanfaatkan basis pengetahuan yang luas, LLM dapat merancang prosedur pengujian inovatif untuk mengungkap kerentanan potensial.

Keterbatasan LLM dalam Pengujian Penetrasi Otomatis (Deng et al., 2024):

a. Kehilangan Konteks Sesi (*Long-Term Memory*): LLM kesulitan mempertahankan pemahaman yang koheren tentang skenario pengujian secara keseluruhan karena batasan token window, yang dapat menyebabkan pengabaian penemuan sebelumnya.
b. Terlalu Menekankan Tugas Terbaru (*Depth-First Search*): LLM cenderung terlalu menekankan tugas-tugas terbaru dalam riwayat percakapan, mengadopsi pendekatan depth-first search, yang dapat menyebabkan pengabaian permukaan serangan potensial lainnya.
c. Generasi Hasil yang Tidak Akurat dan Halusinasi: LLM menunjukkan masalah dengan generasi hasil yang tidak akurat dan halusinasi, seringkali salah mengonfigurasi alat atau bahkan menciptakan alat yang tidak ada.
d. Interpretasi Gambar: LLM saat ini tidak dapat memproses gambar, yang krusial dalam skenario pengujian penetrasi tertentu.

e. Teknik Rekayasa Sosial dan Isyarat Halus: LLM tidak memiliki kemampuan untuk menggunakan teknik rekayasa sosial tertentu dan mendeteksi isyarat halus.
f. Konstruksi Kode Eksploitasi: Meskipun memiliki kemahiran dalam pemahaman dan generasi kode, LLM kurang dalam menghasilkan script eksploitasi yang terperinci.

Dampak Rekayasa Prompt (*Prompt Engineering*) pada Kinerja Deteksi Kerentanan Model GPT

Rekayasa prompt secara signifikan mempengaruhi kinerja model GPT dalam deteksi kerentanan. Perubahan kecil pada prompt dapat secara substansial mempengaruhi kualitas output yang dihasilkan (Bae et al., 2024).

Tabel 2. Pengaruh pada Kinerja Model (Bae et al., 2024)

GPT-4o	Claude-3.5 Sonnet	GPT-3.5 Turbo
Prompt "Step-by-Step" secara signifikan meningkatkan kinerja (F1 score 0.9072), mengurangi false positive dengan mendorong analisis terstruktur. Prompt "Concise" (AUC 0.74) berguna untuk meminimalkan false positive dan negative.	Prompt "Step-by-Step" juga menunjukkan kinerja terbaik (F1 score 0.8933, AUC 0.74), membatasi analisis yang tidak perlu.	Penyesuaian prompt memiliki dampak yang dapat diabaikan pada kinerja (F1 score 0.65-0.67, AUC 0.62-0.65), dengan potensi false positive yang tinggi.

Tabel 3. Pengaruh pada Generasi Unit Test (Antal et al., 2025)

Spesifikasi Peran	Stimulasi Emosional	Identifikasi CWE
Menentukan peran (misal, "penguji perangkat lunak senior") meningkatkan kebenaran sintaksis dan semantik.	Menghilangkan daya tarik emosional secara tak terduga meningkatkan kebenaran sintaksis tetapi menurunkan kebenaran semantik.	Menambahkan pengidentifikasi CWE secara tak terduga menurunkan kebenaran sintaksis dan semantik

Model yang lebih canggih (GPT-4o, Claude-3.5 Sonnet) lebih responsif terhadap penyesuaian prompt daripada model yang lebih lama (GPT-3.5 Turbo) (Bae et al., 2024). Ini menunjukkan bahwa kinerja LLM tidak hanya bergantung pada arsitektur model, tetapi juga pada cara interaksi dengan model tersebut (Antal et al., 2025).

Analisis Komparatif: GPT vs. Burp Suite dalam Pengujian Kerentanan Aplikasi Web

Perbandingan antara Burp Suite dan model GPT menyoroti perbedaan mendasar dalam pendekatan dan spesialisasi mereka. Burp Suite adalah alat khusus, dirancang sebagai *all-in-one toolkit* yang sangat dioptimalkan untuk pengujian penetrasi aplikasi web, dengan pendekatan berbasis aturan, pola, dan interaksi HTTP yang terstruktur (Mathew dan Benjamin, 2021; Choudhary et al., 2023). Sebaliknya, model GPT adalah model bahasa umum yang diterapkan pada tugas keamanan siber, dengan kemampuan yang berasal dari pembelajaran pola dari data teks dan kode yang luas, dilengkapi dengan kemampuan penalaran dan generasi (Deng et al., 2024).

Analisis Komparatif: GPT vs. Burp Suite dalam Pengujian Kerentanan Aplikasi Web

Perbandingan antara Burp Suite dan model GPT menyoroti perbedaan mendasar dalam pendekatan dan spesialisasi mereka. Burp Suite adalah alat khusus, dirancang sebagai *all-in-one toolkit* yang sangat dioptimalkan untuk pengujian penetrasi aplikasi web, dengan pendekatan berbasis aturan, pola, dan interaksi HTTP yang terstruktur (Mathew dan Benjamin, 2021; Choudhary et al., 2023). Sebaliknya, model GPT adalah model bahasa umum yang diterapkan pada tugas keamanan siber, dengan kemampuan yang berasal dari pembelajaran pola dari data teks dan kode yang luas, dilengkapi dengan kemampuan penalaran dan generasi (Deng et al., 2024).

Tabel 4. Analisis Komparatif Burp Suite vs. Model GPT

Kriteria Perbandingan	Burp Suite	Model GPT
Pendekatan Utama	Alat khusus, berbasis aturan dan interaksi HTTP (Mathew dan Benjamin, 2021; Choudhary et al., 2023)	Model bahasa umum, berbasis pembelajaran pola dari teks/kode, kemampuan penalaran dan generasi (Deng et al., 2024)
Spesialisasi	Sangat dioptimalkan untuk pengujian aplikasi web (Mathew dan Benjamin, 2021)	Generalis, dapat diterapkan pada berbagai tugas keamanan siber (Deng et al., 2024)
Keunggulan Utama	Kedalaman fungsionalitas, akurasi teruji (misal,	Otomatisasi tugas berulang, analisis kode & generasi *shell*,

	misconfiguration), kontrol manual, kematangan alat, penanganan sesi yang efektif (Mathew dan Benjamin, 2021; Jarupunphol et al., 2023)	pemahaman konteks luas (dalam batasan *token*), adaptasi cepat, generasi *output* inovatif (Deng et al., 2024; Antal et al., 2025)
Keterbatasan Utama	Tidak selalu komprehensif (misal, XSS lemah), potensi *false positive* (Mathew dan Benjamin, 2021; Jarupunphol et al., 2023)	Kehilangan konteks sesi, halusinasi & ketidakakuratan, tingkat *false positive*/*negative* tinggi, bias *false positive*, keterbatasan interpretasi gambar/rekayasa sosial, risiko keamanan (Pelofske et al., 2024; Deng et al., 2024)
Tingkat Otomatisasi	Otomatisasi sebagian (Scanner), kontrol manual yang kuat (Mathew dan Benjamin, 2021)	Potensi otomatisasi tinggi untuk sub-tugas, namun otomatisasi penuh masih terbatas (Deng et al., 2024)
Kebutuhan Intervensi Manusia	Penting untuk pengujian mendalam dan verifikasi hasil	Krusial untuk verifikasi *output* (mitigasi halusinasi/*false positive*) dan perencanaan strategis (Deng et al., 2024)
Biaya/Sumber Daya	Beragam edisi (gratis hingga berbayar mahal) (Mathew dan Benjamin, 2021)	Biaya API (untuk model komersial), sumber daya komputasi tinggi untuk model *open-source* (Bae et al., 2024; Deng et al., 2024)

Sinergi dan Arah Masa Depan

Mengingat kekuatan dan kelemahan yang saling melengkapi dari Burp Suite dan model GPT, pendekatan hibrida yang mengintegrasikan kedua teknologi ini dapat menjadi solusi paling

efektif untuk pengujian kerentanan aplikasi web (Jarupunphol et al., 2023). Model hibrida optimal dapat memanfaatkan Burp Suite untuk menangani pengujian presisi dan interaksi HTTP yang kompleks, sementara LLM dapat memberikan analisis kode skala besar, deteksi anomali, dan bantuan dalam generasi exploit awal atau unit test. LLM dapat bertindak sebagai "asisten cerdas" yang membantu dalam analisis log dan output, generasi *payload* yang cerdas, penjelasan kerentanan, dan pembuatan unit test (Deng et al., 2024; Antal et al., 2025).

Untuk organisasi yang mempertimbangkan integrasi LLM, direkomendasikan adopsi bertahap dan terkontrol, investasi dalam *prompt engineering*, verifikasi manusia berkelanjutan terhadap *output* LLM, dan kombinasi alat (Bae et al., 2024).

Arah penelitian masa depan dalam penerapan AI untuk keamanan siber, khususnya dalam pengujian kerentanan, meliputi peningkatan konteks jangka panjang LLM, mitigasi halusinasi dan bias, pengembangan LLM multimodal, pengembangan dataset yang lebih baik dan *real-time*, AI yang dapat dijelaskan (XAI), serta sistem hibrida AI-manusia (Kaur et al., 2023).

KESIMPULAN DAN SARAN

Berdasarkan hasil analisis literatur yang telah dilakukan, dapat disimpulkan bahwa Burp Suite dan model GPT (*Generative Pre-trained Transformer*) memiliki keunggulan masing-masing dalam pengujian kerentanan aplikasi web. Burp Suite merupakan alat pengujian penetrasi yang komprehensif dan telah teruji, dengan kapabilitas tinggi dalam mendeteksi kerentanan runtime melalui simulasi interaksi pengguna dan analisis untuk HTTP atau *web traffic*. Alat ini mendukung berbagai modul seperti *scanner, intruder, dan repeater* yang sangat berguna dalam pengujian dinamis. Sementara itu, GPT unggul dalam analisis kode sumber secara statis, mampu menghasilkan deteksi kerentanan awal, memberikan saran perbaikan, dan mendukung otomasi sub-tugas teknis seperti analisis shell dan unit test. Model GPT-4o dan Claude-3.5 Sonnet bahkan menunjukkan performa tinggi dalam deteksi kerentanan dengan F1-score di atas 0,90. Namun, model ini masih menghadapi tantangan seperti *false positives*, kehilangan konteks, dan potensi halusinasi hasil.

Melalui perbandingan yang telah dilakukan, pendekatan berbasis Burp Suite dan GPT dinilai saling melengkapi dan berpotensi besar jika digunakan secara terintegrasi. GPT dapat digunakan pada tahap awal untuk menganalisis kode, merancang skenario pengujian, serta mengidentifikasi pola kerentanan berdasarkan pembelajaran mesin, sementara Burp Suite dapat digunakan untuk memvalidasi temuan tersebut melalui pengujian runtime yang terstruktur dan dapat diobservasi langsung. Oleh karena itu, disarankan agar praktisi keamanan dan pengembang aplikasi mempertimbangkan pendekatan hibrida dalam pengujian kerentanan.

Selain itu, pemanfaatan GPT hendaknya disertai dengan teknik rekayasa prompt yang tepat dan validasi hasil oleh tenaga ahli untuk meminimalkan risiko kesalahan interpretasi. Integrasi alat tradisional dengan teknologi berbasis AI dapat menjadi langkah strategis untuk membangun sistem keamanan aplikasi web yang lebih adaptif, efisien, dan cerdas.

DAFTAR PUSTAKA

Antal, G., Bán, D., Isztin, M., Ferenc, R., & Hegedűs, P. (n.d.). Leveraging GPT-4 for Vulnerability-Witnessing Unit Test Generation.

Bae, J., Kwon, S., & Myeong, S. (2024). Enhancing Software Code Vulnerability Detection Using GPT-4o and Claude-3.5 Sonnet: A Study on Prompt Engineering Techniques. Electronics, 13(13), 2657.

Deng, G., Liu, Y., Mayoral-Vilches, V., Liu, P., Li, Y., Xu, Y., … Rass, S. (2024). PENTESTGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing.

Fu, M., Chakkrit Kla Tantithamthavorn, Nguyen, V., & Le, T. (2023). ChatGPT for Vulnerability Detection, Classification, and Repair: How Far Are We? ArXiv (Cornell University).

Hadi, M. A., Najm Abdulredha, M., & Hasan, E. (2023). Introduction to ChatGPT: A new revolution of artificial intelligence with machine learning algorithms and cybersecurity. Science Archives, 04(04), 276–285.

Kaur, R., Gabrijelčič, D., & Klobučar, T. (2023). Artificial intelligence for cybersecurity: Literature review and future research directions. Information Fusion, 97(101804), 1–29. ScienceDirect.

Khare, A., Dutta, S., Li, Z., Solko-Breslin, A., Alur, R., & Naik, M. (2023). Understanding the Effectiveness of Large Language Models in Detecting Security Vulnerabilities. ArXiv (Cornell University).

Ms. Jetty Banjamin, & Dona Rose Mathew. (2022). Penetration Testing and Vulnerability Scanning of Web Application Using Burp Suite. *National Conference on Emerging Computer Applications*, *3*(1). Retrieved from https://ajcejournal.in/nceca/article/view/108

Pelofske, E., Urias, V., & Liebrock, L. M. (2024). Automated Software Vulnerability Static Code Analysis Using Generative Pre-Trained Transformer Models. ArXiv (Cornell University).

Pita Jarupunphol, Suppachochai Seatun, & Wipawan Buathong. (2023). Measuring Vulnerability Assessment Tools' Performance on the University Web Application. Pertanika Journal of Science and Technology, 31(6), 2973–2993.

Rhythm Choudhary, Jhanvii Rawat, & Garima Singh. (2023). Comprehensive Exploration of Web Application Security Testing with Burp Suite Tools. International Journal for Multidisciplinary Research, 5(6).