Jurnal JAMIKA Manajemen Informatika

Jurnal Manajemen Informatika (JAMIKA)
Volume xx Nomor x Edisi xxxx 20xx
E ISSN: 2655-6960 | P ISSN: 2088-4125
OJS: https://ojs.unikom.ac.id/index.php/jamika

Implementasi Otomatisasi Deteksi Phishing Multimodal Berbasis Orkestrasi Workflow n8n dan Large Language Model

Multimodal Phishing Detection Automation Based on n8n Workflow Orchestration and Large Language Models

Muhammad Rizki Adha[1], **Muhammad Idris**[2]
Program Studi Rekayasa Keamanan Siber, Politeknik Negeri Batam, Batam, Indonesia
E-mail: m.rizki@students.polibatam.ac.id

Abstrak

Peningkatan serangan phishing menuntut sistem deteksi yang dapat bekerja otomatis, terstruktur, mudah diintegrasikan, dan mampu memanfaatkan lebih dari satu jenis artefak analisis. Penelitian ini bertujuan merancang dan mengimplementasikan purwarupa deteksi phishing multimodal berbasis otomatisasi workflow menggunakan n8n pada skala Proof of Concept. Sistem menerima input berupa URL*, kemudian menjalankan pemeriksaan reputasi melalui VirusTotal, pengambilan artefak teknis, pengambilan tangkapan layar halaman web, analisis visual, analisis teks, dan klasifikasi akhir. Artefak teknis meliputi data WHOIS, sertifikat TLS, security headers, serta struktur DOM/HTML. Analisis visual dilakukan menggunakan Llama 3.2 Vision 11B melalui Ollama, sedangkan hasil VirusTotal, artefak teknis, dan analisis visual diproses menggunakan Gemma-3 12B-IT-QAT untuk menghasilkan deteksi berupa klasifikasi, skor risiko, dan alasan keputusan dalam format JSON. Pengujian dilakukan terhadap 200* URL*, terdiri atas 160 benign* URL *dan 40 phishing* URL*. Hasil pengujian menunjukkan accuracy 93,50%, precision 82,93%, recall 85,00%, dan F1-score 83,95%. Hasil ini menunjukkan bahwa n8n dapat digunakan sebagai orkestrator workflow untuk mendukung analisis phishing multimodal pada skala Proof of Concept. Namun, sistem masih lebih tepat digunakan sebagai alat bantu analisis awal karena latensi dan performanya dipengaruhi keterbatasan resource perangkat, sehingga belum merepresentasikan sistem produksi, dan memerlukan pengujian lanjutan pada data lebih besar serta lingkungan operasional nyata.*

Kata kunci: *Deteksi Phishing, n8n, Workflow, VirusTotal, Ollama.*

Abstract

The increasing number of phishing attacks requires an automatic detection system that can integrate multiple analytical artifacts. This study aims to design and implement a multimodal phishing detection prototype based on workflow automation using n8n at Proof of Concept scale. The system receives a URL *and performs reputation checking through VirusTotal, technical artifact collection, webpage screenshot capture, visual analysis, text analysis, and final classification. The technical artifacts include WHOIS data, TLS certificates, security headers, and DOM/HTML structure. Visual analysis is performed using Llama 3.2 Vision 11B through Ollama, while VirusTotal results, technical artifacts, and visual outputs are processed using Gemma-3 12B-IT-QAT to generate a detection JSON output containing classification, risk score, and decision rationale. The system was tested using 200* URL*s consisting of 160 benign* URL*s and 40 phishing* URL*s. The results show an accuracy of 93.50%, precision of 82.93%, recall of 85.00%, and F1-score of 83.95%. These results indicate that n8n can be used as a workflow orchestrator to support multimodal phishing analysis at Proof of Concept scale. However, the system is more appropriate as an analysis support tool because its latency and performance are affected by device resource limitations, so it requires further testing on larger datasets.*

Keywords: *Phishing Detection, n8n, Workflow, VirusTotal, Ollama.*

Naskah diterima xx Jan. 2024; direvisi xx Feb. 2024; dipublikasikan xx Apr. 2024.

I. PENDAHULUAN

Ancaman *phishing* terus meningkat seiring berkembangnya teknik rekayasa sosial dan penyamaran halaman web. Laporan Anti-*Phishing* Working Group (APWG) pada kuartal pertama tahun 2025 mencatat lebih dari 1.003.924 serangan *phishing* unik dalam satu kuartal [1]. Di Indonesia, laporan *Indonesia Domain Abuse Data Exchange* (IDADX) juga mencatat 85.414 kasus penyalahgunaan domain pada kuartal IV tahun 2024 [2]. Data tersebut mengindikasikan bahwa proses pemeriksaan URL mencurigakan tidak cukup jika hanya dilakukan secara manual, terutama karena *phishing* URL sering aktif dalam waktu singkat sebelum dinonaktifkan [1], [3].

Phishing modern tidak hanya menggunakan URL yang menyerupai domain asli, tetapi juga memanfaatkan tampilan visual yang mirip dengan situs resmi. Aktor ancaman menyembunyikan elemen penting, seperti logo, formulir *login*, dan teks tertentu, dalam bentuk gambar agar sulit dibaca oleh pemindai berbasis teks. Pola ini membuat analisis visual terhadap halaman web menjadi penting dalam deteksi *phishing* [3], [4]. Kondisi tersebut membuat proses deteksi perlu memeriksa lebih dari satu jenis data, seperti URL, struktur halaman, sertifikat TLS, *security headers*, dan *screenshot* halaman web [3], [5].

Beberapa penelitian sebelumnya telah mengembangkan sistem deteksi *phishing* menggunakan *machine learning* atau model kecerdasan buatan [3], [6], [7]. Namun, sebagian sistem masih berfokus pada pengujian model, bukan pada implementasi alur kerja yang dapat digunakan secara langsung dalam proses analisis. Selain itu, penggunaan layanan berbasis *cloud* juga dapat menimbulkan ketergantungan pada pihak ketiga, terutama ketika data yang dianalisis bersifat sensitif [8]. Oleh karena itu, dibutuhkan sistem yang tidak hanya mampu mendeteksi *phishing*, tetapi juga mampu mengotomatisasi proses pengumpulan data, analisis, dan penyajian hasil secara terstruktur.

Penelitian ini mengusulkan purwarupa sistem deteksi *phishing* berbasis otomatisasi *workflow* menggunakan n8n [9]. Platform n8n digunakan sebagai orkestrator utama yang mengelola alur deteksi mulai dari penerimaan URL, pemeriksaan reputasi melalui VirusTotal, pengambilan artefak teknis, pengambilan *screenshot* halaman web, pemanggilan model AI lokal, hingga pembentukan keluaran deteksi berformat JSON. Eksekusi paralel pada n8n dimanfaatkan untuk mengatur beberapa proses pengumpulan artefak agar berjalan dalam alur yang lebih terstruktur, bukan untuk mengoptimalkan kecepatan inferensi model AI. Pemeriksaan reputasi awal memanfaatkan API publik VirusTotal, sedangkan analisis konten web berupa *screenshot* dan struktur HTML dijalankan secara lokal melalui Ollama [10]. Dengan pendekatan ini, n8n berperan sebagai penghubung dan pengendali proses analisis, sementara kinerja inferensi tetap dipengaruhi oleh model yang digunakan dan keterbatasan *resource* perangkat.

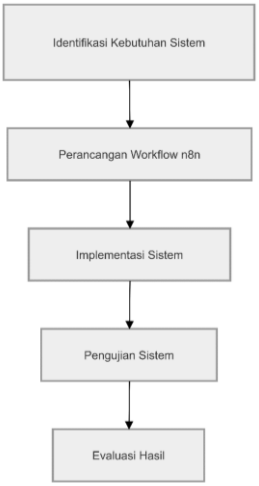
Sistem yang dikembangkan dirancang untuk menjalankan proses deteksi *phishing* secara otomatis melalui *workflow* n8n. Sistem menerima *input* berupa URL, kemudian melakukan pemeriksaan reputasi URL melalui VirusTotal, mengambil artefak teknis, mengambil *screenshot* halaman, menjalankan analisis visual, dan menghasilkan klasifikasi akhir. Artefak teknis yang digunakan meliputi data WHOIS, sertifikat TLS, *security headers*, dan struktur DOM/HTML. Tangkapan layar halaman dianalisis menggunakan Llama 3.2 Vision 11B melalui Ollama [10], [11], sedangkan hasil VirusTotal, artefak teknis, dan hasil analisis visual diproses menggunakan Gemma-3 12B-IT-QAT [12], [13] untuk menghasilkan keluaran deteksi berformat JSON berupa klasifikasi, skor risiko, dan alasan keputusan. Proses inferensi dijalankan secara lokal melalui Ollama agar data hasil pemeriksaan tetap berada pada perangkat pengujian.

Penelitian ini bertujuan untuk merancang dan mengimplementasikan purwarupa sistem deteksi *phishing* multimodal berbasis otomatisasi *workflow* menggunakan n8n, serta mengevaluasi kelayakan awal sistem pada skala *Proof of Concept* (PoC). Evaluasi dibatasi pada 200 sampel URL yang terdiri atas 160 *benign* URL dan 40 *phishing* URL, dengan penilaian utama berdasarkan hasil klasifikasi dan waktu respons sistem. Batasan ini menunjukkan bahwa penelitian tidak ditujukan untuk melatih model baru, menguji skalabilitas produksi, atau menggantikan peran analis keamanan, melainkan untuk menilai apakah orkestrasi *workflow* n8n dapat mengintegrasikan pemeriksaan reputasi URL, pengambilan artefak teknis, analisis visual, analisis teks, dan pembentukan keluaran deteksi berformat JSON dalam satu alur otomatis. Dengan demikian, kontribusi penelitian ini terletak pada implementasi dan evaluasi awal alur deteksi *phishing* multimodal yang terotomatisasi, bukan pada klaim performa sistem dalam lingkungan operasional berskala besar.

II. METODE PENELITIAN

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak terapan untuk merancang, mengimplementasikan, dan menguji purwarupa sistem deteksi *phishing* berbasis otomatisasi *workflow* menggunakan n8n. Pendekatan ini dipilih karena fokus penelitian bukan pada pelatihan model baru, melainkan pada pembangunan alur otomatis yang dapat menerima URL, mengambil artefak teknis, menjalankan pemeriksaan OSINT, melakukan analisis AI lokal, dan menghasilkan klasifikasi akhir secara otomatis.

Tahapan penelitian terdiri atas lima tahap utama, yaitu identifikasi kebutuhan sistem, perancangan *workflow*, implementasi sistem, pengujian sistem, dan evaluasi hasil. Tahapan tersebut ditunjukkan pada Gambar 1.

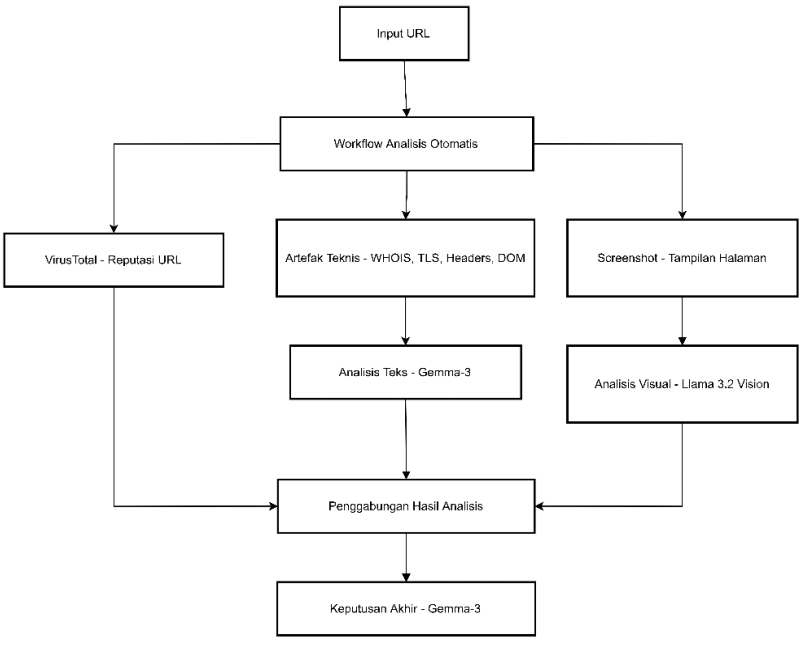


Gambar 1. Metode Penelitian

Secara umum, sistem dirancang untuk menerima URL melalui *webhook* n8n. Setelah URL diterima, sistem menjalankan beberapa proses pemeriksaan, yaitu pemeriksaan reputasi URL melalui OSINT, pengambilan data WHOIS, pemeriksaan sertifikat TLS, pemeriksaan *security headers*, pengambilan struktur halaman, dan pengambilan *screenshot* halaman web. Hasil dari setiap proses dikumpulkan dalam format *JavaScript Object Notation* (JSON), kemudian digunakan sebagai dasar klasifikasi akhir.

Rancangan sistem dibuat sebagai alur analisis otomatis tanpa pemilahan jalur berdasarkan ambang deteksi vendor. Setiap URL yang masuk diproses melalui tahapan yang sama, yaitu standardisasi URL, pemeriksaan reputasi melalui VirusTotal, pengambilan artefak teknis, pengambilan *screenshot* halaman web, analisis visual, analisis teks, dan pembentukan keputusan akhir. Dalam rancangan ini, n8n berperan sebagai orkestrator yang menghubungkan beberapa layanan dan model AI lokal dalam satu *workflow*. Hasil akhir sistem disajikan sebagai keluaran deteksi berformat JSON yang memuat status klasifikasi URL, skor risiko, dan alasan keputusan.

Dalam penelitian ini, keluaran deteksi berformat JSON merujuk pada hasil akhir yang dikembalikan sistem setelah seluruh proses analisis selesai. Adapun data WHOIS, sertifikat TLS, *security headers*, struktur DOM/HTML, *screenshot* halaman web, hasil analisis visual, dan hasil analisis teks diposisikan sebagai artefak atau keluaran antara yang digunakan untuk mendukung pembentukan keputusan akhir.



Gambar 2. Arsitektur Workflow Deteksi *Phishing* Menggunakan n8n

Lingkungan pengujian disiapkan secara lokal atau *on-premise* untuk menjaga data hasil pemeriksaan tetap berada pada perangkat peneliti. Konfigurasi ini juga digunakan untuk menguji apakah sistem masih dapat berjalan dengan perangkat keras terbatas, terutama pada penggunaan model AI lokal.

TABEL 1
LINGKUNGAN EKSPERIMEN

No	Komponen	Spesifikasi / Keterangan	Fungsi
1	Prosesor	Intel Core i7-14700	Menjalankan n8n dan proses pendukung
2	RAM	24 GB DDR4	Mendukung eksekusi *workflow* dan proses AI lokal
3	Penyimpanan	SSD NVMe 512 GB	Menyimpan log, hasil uji dan artefak sistem
4	GPU	NVIDIA GeForce RTX 5070 Laptop, 8 GB VRAM	Menjalankan inferensi model AI lokal
5	Platform otomasi	n8n versi 2.17.6	Mengatur alur otomatisasi deteksi
6	Mesin AI lokal	Ollama versi 0.20.0	Menjalankan model AI lokal
7	Model visual	Llama 3.2 Vision 11B	Menganalisis *screenshot* halaman
8	Model teks dan aggregator	Gemma-3 12B-IT-QAT	Menganalisis artefak teks dan menghasilkan klasifikasi akhir

Implementasi sistem dilakukan menggunakan beberapa *node* dan *sub-workflow* pada n8n. Setiap *node* memiliki fungsi khusus agar proses deteksi dapat berjalan secara modular. Ringkasan *node* utama yang digunakan ditunjukkan pada Tabel 2.

TABEL 2
NODE DAN SUB-*WORKFLOW* N8N

No	Komponen *Workflow / Node*	Fungsi	Input	*Output*
1	*Main Webhook Trigger*	Menerima *payload* dari pengguna sebagai titik masuk utama sistem.	URL target	URL tervalidasi
2	*Node Execute Analysis Workflow in Parallel*	Menjalankan beberapa proses pengumpulan artefak secara paralel agar alur analisis lebih terstruktur.	URL tervalidasi	Kumpulan artefak teknis awal
3	*Sub-workflow VirusTotal Scan*	Memeriksa reputasi URL melalui API VirusTotal	URL target	Data reputasi URL
4	*Sub-workflow WHOIS Lookup*	Mengambil data registrasi domain, seperti umur domain dan registrar.	Domain target	Data WHOIS berformat JSON
5	*Sub-workflow Scan SSL*	Mengekstrak dan memvalidasi informasi sertifikat TLS/SSL.	Domain target	Data sertifikat TLS/SSL
6	*Sub-workflow Scan Security Headers*	Memeriksa kebijakan keamanan HTTP, seperti CSP, HSTS, dan header keamanan lainnya.	URL target	Status *security headers*
7	*Sub-workflow Scrape Website*	Mengekstrak struktur DOM/HTML halaman web secara ringkas.	URL target	Data DOM/HTML
8	*Sub-workflow Screenshot Page*	Mengambil *screenshot* halaman web untuk analisis visual.	URL target	Screenshot halaman web
9	*Sub-workflow Trigger Visual Analysis*	Memproses *screenshot* halaman web menggunakan model Llama 3.2 Vision.	*Screenshot* halaman web	Hasil analisis visual
10	*Sub-workflow Trigger Text Analysis*	Memproses artefak teknis dan hasil pemeriksaan reputasi menggunakan model Gemma-3	Artefak teknis dan data reputasi URL	Hasil analisis teks

11	*Sub-workflow Trigger Aggregation and Decision*	Mengagregasi hasil analisis visual dan hasil analisis teks untuk menghasilkan keputusan klasifikasi akhir.	Hasil analisis visual dan hasil analisis teks	Klasifikasi akhir, skor risiko, dan alasan keputusan
12	*Node Log to Sheets*	Mencatat hasil klasifikasi, waktu respons, dan data evaluasi sistem ke *Google Sheets.*	Keluaran akhir sistem	Log evaluasi sistem
13	*Node Respond Webhook (Full Scan)*	Mengembalikan hasil analisis kepada klien melalui respons HTTP.	Keluaran deteksi berformat JSON	Respons HTTP 200 berisi keluaran deteksi berformat JSON

Dalam penelitian ini, istilah *phishing* URL digunakan untuk merujuk pada URL yang berasal dari sumber umpan ancaman dan memiliki indikasi sebagai situs phishing. Istilah *benign* URL digunakan untuk merujuk pada URL aman atau non-*phishing* dalam data uji, yaitu URL yang tidak diklasifikasikan sebagai *phishing* berdasarkan sumber data dan proses evaluasi sistem. Pada penelitian ini, *benign* URL diperoleh dari daftar situs populer Tranco dan digunakan sebagai pembanding terhadap *phishing* URL

Pengujian dilakukan menggunakan 200 sampel URL. Sampel tersebut terdiri atas 40 *phishing* URL aktif dan 160 *benign* URL. *Phishing* URL diperoleh dari sumber umpan ancaman publik seperti PhishTank dan OpenPhish [14], [15], sedangkan *benign* URL diperoleh dari daftar situs populer Tranco [16]. Komposisi data uji ditunjukkan pada Tabel 3.

Jumlah 200 sampel URL dipilih karena penelitian ini berfokus pada pengujian kelayakan purwarupa berbasis *Proof of Concept* (PoC), bukan pada pelatihan model atau pengujian statistik berskala besar. Setiap URL diproses melalui pemeriksaan reputasi, pengambilan artefak teknis, *screenshot*, analisis visual, analisis teks, dan klasifikasi akhir. Oleh karena itu, jumlah tersebut digunakan untuk menilai fungsi sistem, stabilitas *workflow*, dan kinerja awal sistem. Pengujian dengan jumlah URL yang lebih besar dapat dilakukan pada penelitian lanjutan untuk menilai skalabilitas sistem pada kondisi operasional yang lebih luas.

TABEL 3
DISTRIBUSI DATASET URL

No	Kategori URL	Sumber Data	Jumlah Spesimen
1	*Phishing* URL	PhishTank dan OpenPhish	40
2	*Benign* URL	Tranco	160
	Total Spesimen		200

Skenario pengujian disusun untuk memastikan sistem dapat memproses berbagai kondisi URL. Pengujian tidak hanya menilai hasil klasifikasi, tetapi juga melihat apakah *workflow* n8n dapat berjalan sesuai alur yang dirancang. Skenario pengujian ditunjukkan pada Tabel 4.

TABEL 4
SKENARIO PENGUJIAN SISTEM

No	Skenario Pengujian	Input	*Output* yang diharapkan	Indikator
1	Pemeriksaan *Phishing* URL	*Phishing* URL aktif	Sistem menghasilkan klasifikasi *phishing*	Klasifikasi sesuai label aktual
2	Pemeriksaan *Benign* URL	URL dari Tranco	Sistem menghasilkan klasifikasi *benign*	Klasifikasi sesuai label aktual
3	Pemeriksaan reputasi URL	URL Target	VirusTotal menghasilkan data reputasi URL	Data reputasi URL terbentuk
4	Pengambilan artefak teknis	URL target	*WHOIS, TLS, security headers,* dan *DOM/HTML* berhasil diperoleh	Artefak teknis terbentuk
5	Analisis visual halaman	*Screenshot* halaman web	Llama 3.2 Vision menghasilkan analisis visual	*Output* analisis visual terbentuk
6	Pembentukan keputusan akhir	Hasil VirusTotal, artefak teknis, dan hasil analisis visual	Gemma-3 menghasilkan keluaran deteksi berformat JSON	JSON berisi klasifikasi, skor risiko, dan alasan

Evaluasi sistem dilakukan berdasarkan dua aspek, yaitu kinerja klasifikasi dan kinerja eksekusi *workflow*. Kinerja klasifikasi diukur menggunakan *accuracy*, *precision*, *recall*, dan *F1-score*. Metrik ini digunakan untuk melihat kemampuan sistem dalam membedakan *phishing* URL dan *benign* URL. Sementara itu, kinerja eksekusi *workflow* diukur berdasarkan keberhasilan proses otomatisasi pada n8n, terbentuknya artefak teknis, terbentuknya hasil analisis visual, serta keluaran JSON dari Gemma-3.

TABEL 5
METRIK EVALUASI

No	Metrik	Definisi	Tujuan Pengukuran
1	*Accuracy*	Mengukur proporsi seluruh prediksi yang benar terhadap seluruh data uji.	Menilai tingkat ketepatan sistem secara keseluruhan dalam mengklasifikasikan *phishing* URL dan *benign* URL.
2	Precision	Mengukur proporsi URL yang diprediksi sebagai *phishing* dan benar-benar merupakan *phishing* URL.	Menilai ketepatan sistem ketika memberikan prediksi *phishing*, terutama untuk melihat seberapa kecil kesalahan dalam menandai *benign* URL sebagai *phishing*.
3	*Recall*	Mengukur proporsi *phishing* URL aktual yang berhasil terdeteksi sebagai *phishing* oleh sistem.	Menilai kemampuan sistem dalam menemukan *phishing* URL dari seluruh *phishing* URL yang terdapat pada data uji.
4	*F1-score*	Mengukur keseimbangan antara *precision* dan *recall*.	Menilai performa sistem secara lebih seimbang ketika terdapat perbedaan antara ketepatan prediksi *phishing* dan kemampuan menemukan *phishing* URL.
5	Latensi	Mengukur waktu yang dibutuhkan sistem untuk memproses satu URL sejak *input* diterima hingga keluaran deteksi dihasilkan.	Menilai efisiensi waktu pemrosesan sistem pada skala *Proof of Concept.*
6	*False Positive Rate*	Mengukur proporsi *benign* URL yang salah diklasifikasikan sebagai *phishing*	Menilai tingkat kesalahan sistem dalam memberikan peringatan *phishing* pada *benign* URL.

Accuracy dihitung dari proporsi prediksi benar terhadap seluruh data uji. *Precision* dihitung dari perbandingan antara *true positive* dan seluruh prediksi *phishing*, sedangkan *recall* dihitung dari perbandingan antara *true positive* dan seluruh *phishing* URL aktual. F1-*score* digunakan untuk melihat keseimbangan antara *precision* dan *recall. False positive rate* digunakan untuk melihat proporsi *benign* URL yang keliru diklasifikasikan sebagai *phishing*.

Perhitungan metrik klasifikasi dilakukan menggunakan *confusion matrix* yang terdiri atas *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN). *True positive* menunjukkan jumlah *phishing* URL yang diklasifikasikan dengan benar sebagai *phishing*, sedangkan *false negative* menunjukkan *phishing* URL yang salah diklasifikasikan sebagai *benign* URL. *False positive* menunjukkan *benign* URL yang salah diklasifikasikan sebagai *phishing*, sedangkan *true negative* menunjukkan *benign* URL yang diklasifikasikan dengan benar. Berdasarkan komponen tersebut, metrik evaluasi dihitung menggunakan Persamaan (1) sampai Persamaan (5).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

$$FPR = \frac{FP}{FP + TN} \quad (5)$$

Pada penelitian ini, model Llama 3.2 Vision dan Gemma-3 tidak dijalankan secara bersamaan karena keterbatasan VRAM pada perangkat lokal. Kedua model dijalankan secara bergantian melalui Ollama sesuai kebutuhan proses. Kondisi ini menyebabkan waktu analisis menjadi lebih lama, terutama pada tahap analisis visual dan analisis teks. Latensi tinggi tersebut merupakan konsekuensi dari arsitektur eksekusi lokal dan keterbatasan *resource* perangkat, bukan indikasi kegagalan metode deteksi dalam melakukan klasifikasi. Pendekatan ini tetap digunakan karena penelitian berfokus pada pengujian kelayakan purwarupa berbasis *Proof of Concept*, bukan pada optimasi sistem produksi berskala besar.

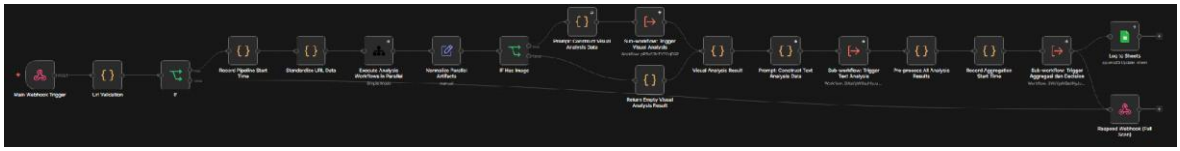
III. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil implementasi dan pengujian purwarupa sistem deteksi *phishing* berbasis otomatisasi *workflow* menggunakan n8n. Pembahasan difokuskan pada implementasi *workflow*, hasil pemeriksaan URL, keluaran sistem, hasil klasifikasi, dan kinerja eksekusi sistem. Fokus pembahasan diarahkan pada bukti bahwa sistem dapat menerima URL, menjalankan proses analisis secara otomatis, dan menghasilkan keluaran deteksi dalam format JSON.

Implementasi Workflow Deteksi Phishing pada n8n

Dalam penelitian ini, sistem deteksi *phishing* diimplementasikan menggunakan *workflow* utama pada n8n. Platform n8n digunakan sebagai orkestrator utama yang mengelola siklus hidup deteksi, mulai dari penerimaan URL, validasi *input*, pengumpulan artefak teknis, analisis visual, analisis teks, hingga pembentukan keputusan akhir. Sistem tidak berjalan secara sekuensial murni. Setelah menerima *input* melalui *node* Main Webhook Trigger dan memvalidasinya, sistem memanfaatkan *node* Execute Analysis Workflows in Parallel untuk menjalankan beberapa proses pengumpulan artefak secara bersamaan. Proses tersebut memicu sub-*workflow* Scan SSL, Scan Security Headers, WHOIS Lookup, VirusTotal Scan, Scrape Website, dan Screenshot Page.

Setelah seluruh artefak terkumpul, *screenshot* halaman web dianalisis menggunakan Llama 3.2 Vision pada sub-*workflow* Trigger Visual Analysis. Pada tahap berikutnya, artefak teknis dan hasil pemeriksaan reputasi dievaluasi melalui sub-*workflow* Trigger Text Analysis menggunakan Gemma-3. Hasil analisis visual dan hasil analisis teks kemudian diagregasi melalui sub-*workflow* Trigger Aggregation and Decision untuk menghasilkan klasifikasi akhir, skor risiko, dan alasan keputusan. Hasil tersebut dicatat melalui *node* Log To Sheets dan dikembalikan kepada pengguna melalui *node* Respond Webhook dalam bentuk keluaran deteksi berformat JSON.



Gambar 3. Implementasi Workflow Utama Deteksi *Phishing* pada n8n

Melalui rancangan ini, proses pemeriksaan URL tidak dilakukan secara manual, tetapi dijalankan secara terstruktur melalui rangkaian *node* dan *sub-workflow* pada n8n. Hal ini menunjukkan bahwa sistem yang dikembangkan tidak hanya berfokus pada hasil klasifikasi, tetapi juga pada implementasi otomasi proses analisis *phishing* dari *input* hingga *output* akhir.

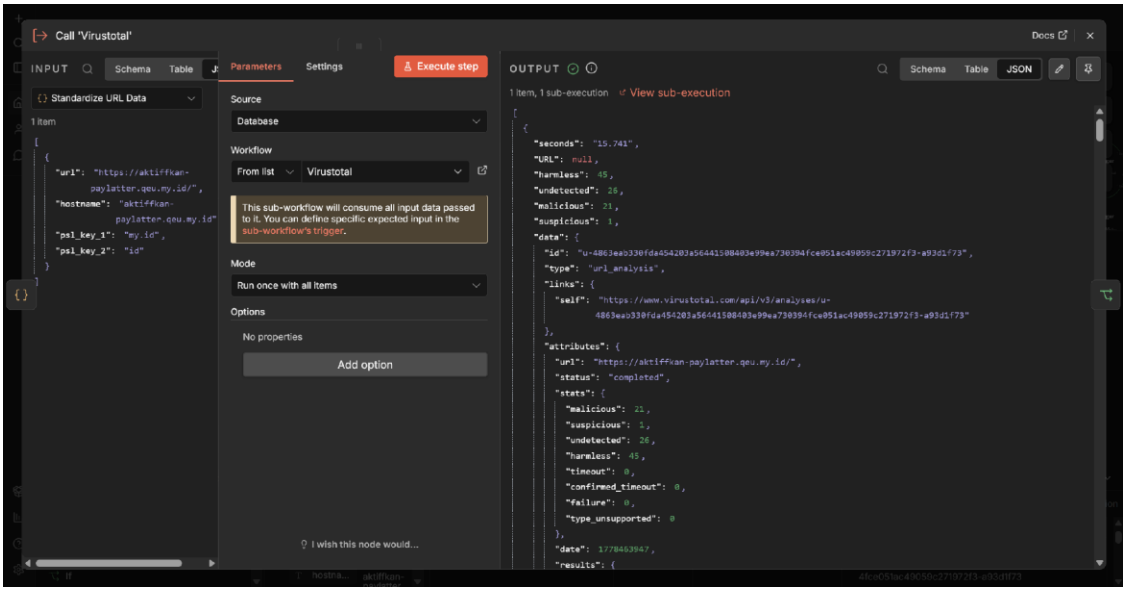
Perbandingan Hasil Pemindaian Phishing URL *dan Benign* URL

Untuk menunjukkan hasil kerja sistem pada *input* aktual, penelitian ini membandingkan hasil pemindaian terhadap satu *phishing* URL dan satu *benign* URL. *Phishing* URL yang digunakan berasal dari domain https://aktiffkan-paylatter.qeu.my.id/, sedangkan *benign* URL yang digunakan adalah https://www.google.com/. Kedua URL diproses melalui alur yang sama, yaitu pemeriksaan reputasi melalui VirusTotal, analisis visual, analisis teks, dan pembentukan keputusan akhir. Perbandingan ini digunakan untuk menunjukkan bagaimana sistem menggabungkan bukti reputasi, visual, dan teknis sebelum menghasilkan klasifikasi akhir.

Pada pengujian *phishing* URL, tahap pertama dilakukan melalui pemeriksaan reputasi menggunakan VirusTotal. Hasil pemeriksaan tersebut ditampilkan pada Gambar 4.

Jurnal JAMIKA Manajemen Informatika

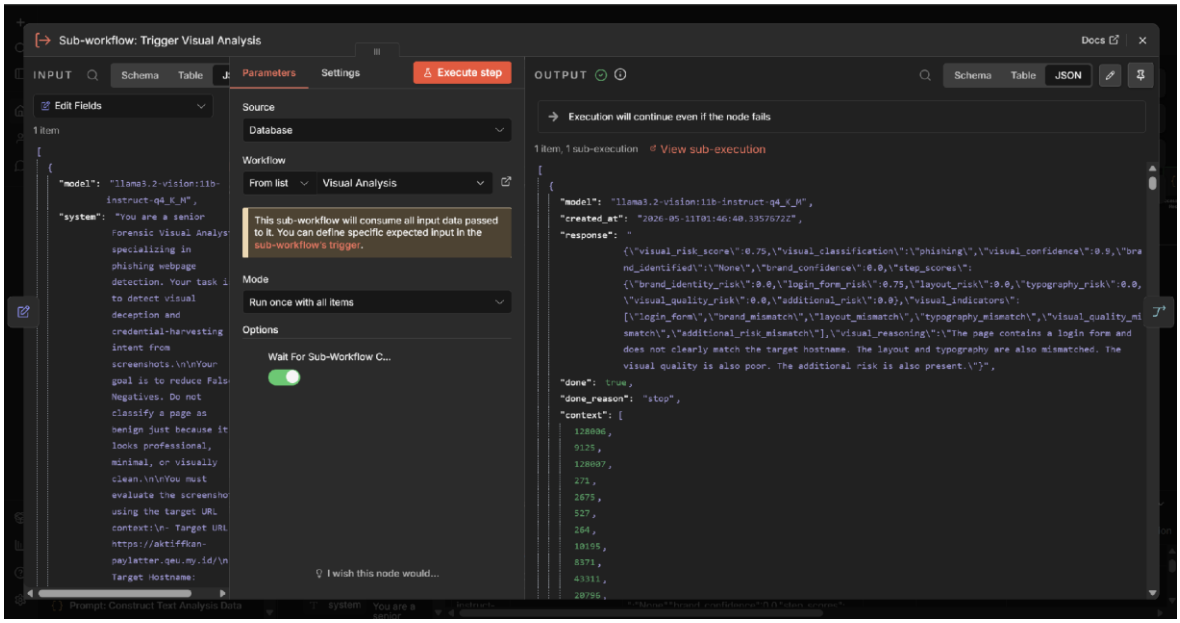
Jurnal Manajemen Informatika (JAMIKA)
Volume xx Nomor x Edisi xxxx 20xx
E ISSN: 2655-6960 | P ISSN: 2088-4125
OJS: https://ojs.unikom.ac.id/index.php/jamika



Gambar 4. Hasil VirusTotal pada *phishing* URL

Berdasarkan Gambar 4, *phishing* URL memperoleh 21 deteksi *malicious* dan 1 deteksi *suspicious*. Hasil ini menunjukkan bahwa URL tersebut telah dikenali oleh sejumlah vendor keamanan sebagai URL berisiko. Data reputasi dari VirusTotal kemudian digunakan sebagai salah satu masukan dalam proses pembentukan keputusan akhir bersama hasil analisis visual dan analisis teks.

Tahap berikutnya adalah analisis visual terhadap tampilan halaman web. Keluaran analisis visual pada *phishing* URL ditampilkan pada Gambar 5.

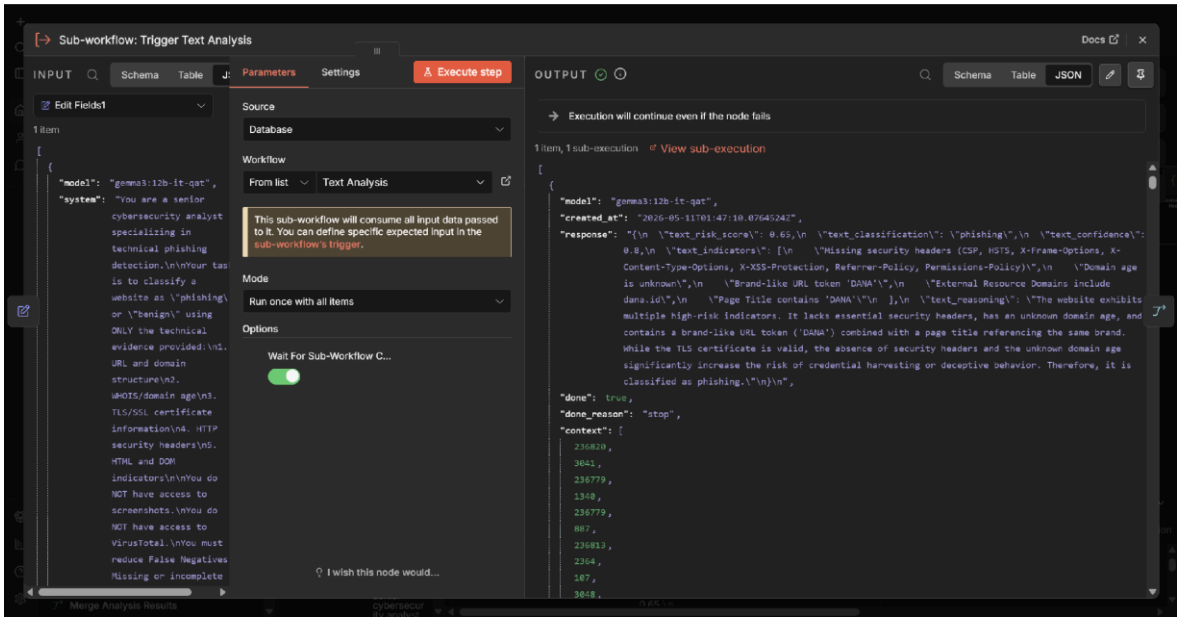


Gambar 5. Hasil Respons JSON *Visual Analysis* pada *phishing* URL

Gambar 5 menunjukkan bahwa sistem menghasilkan visual_risk_score sebesar 0,75 dengan visual_confidence sebesar 0,90. Nilai tersebut mengindikasikan bahwa tampilan halaman memiliki karakteristik visual yang mengarah pada *phishing*. Indikator utama yang mendukung hasil ini meliputi keberadaan formulir *login*, ketidaksesuaian *brand*, perbedaan tata letak, serta kualitas visual yang menyerupai halaman layanan resmi tetapi berada pada domain yang tidak sesuai.

Selain analisis visual, sistem juga melakukan analisis teks terhadap artefak teknis yang diperoleh dari proses pemindaian. Keluaran analisis teks pada *phishing* URL ditampilkan pada Gambar 6.

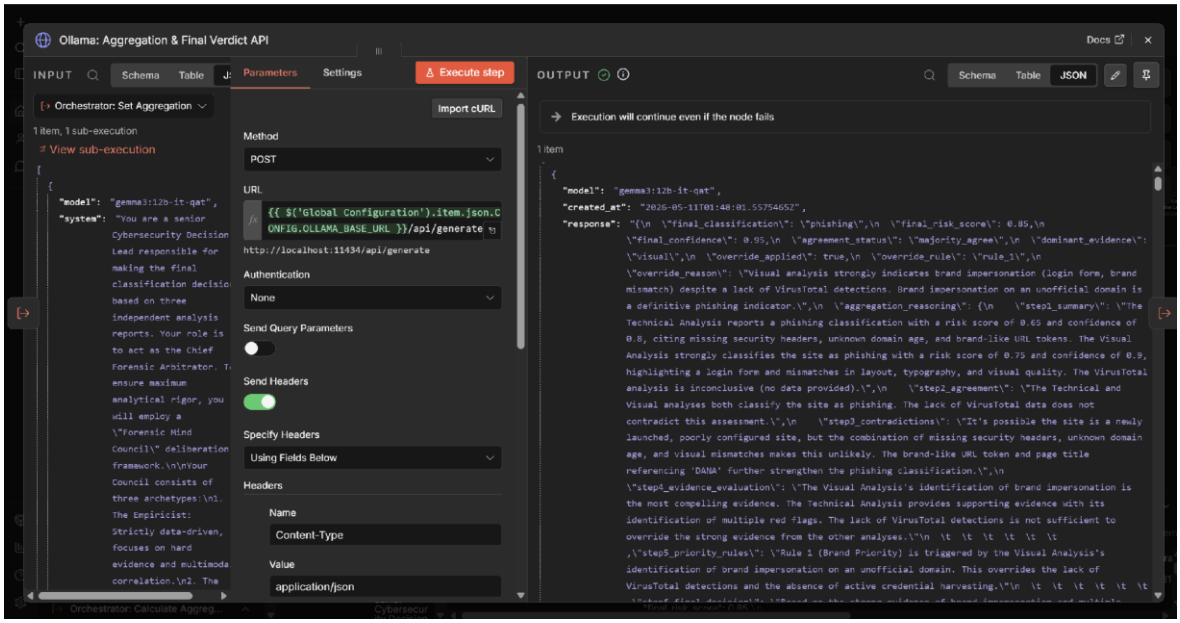
Jurnal JAMIKA Manajemen Informatika
Jurnal Manajemen Informatika (JAMIKA)
Volume xx Nomor x Edisi xxxx 20xx
E ISSN: 2655-6960 | P ISSN: 2088-4125
OJS: https://ojs.unikom.ac.id/index.php/jamika



Gambar 6. Hasil Respons JSON *Text Analysis* pada *phishing* URL

Berdasarkan Gambar 6, sistem menghasilkan klasifikasi *phishing* dengan text_risk_score sebesar 0,65 dan text_confidence sebesar 0,80. Klasifikasi ini didukung oleh beberapa indikator teknis, seperti tidak lengkapnya *security headers*, umur domain yang tidak teridentifikasi, pola token URL yang menyerupai layanan pembayaran, serta elemen halaman yang berupaya menyamarkan identitas. Temuan ini memperkuat hasil analisis visual dan reputasi URL yang sebelumnya telah menunjukkan indikasi risiko.

Setelah hasil VirusTotal, analisis visual, dan analisis teks digabungkan, sistem menghasilkan keputusan akhir sebagaimana ditampilkan pada Gambar 7.



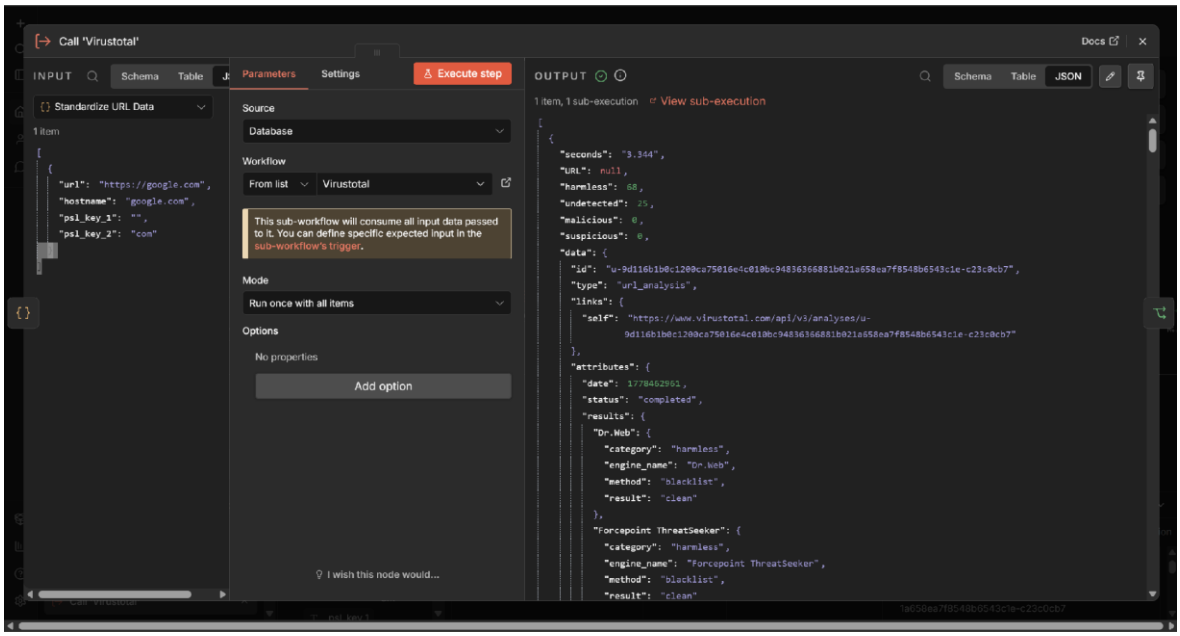
Gambar 7. Respons JSON *Final Decision* pada *phishing* URL

Gambar 7 menunjukkan bahwa sistem mengklasifikasikan URL tersebut sebagai *phishing* dengan final_risk_score sebesar 0,85 dan final_confidence sebesar 0,95. Keputusan akhir ini menunjukkan adanya konsistensi antara bukti reputasi, indikator visual, dan indikator teknis. Dengan demikian, *phishing* URL pada contoh ini berhasil dikenali melalui kombinasi beberapa sumber analisis, bukan hanya berdasarkan satu indikator tunggal.

Jurnal JAMIKA Manajemen Informatika

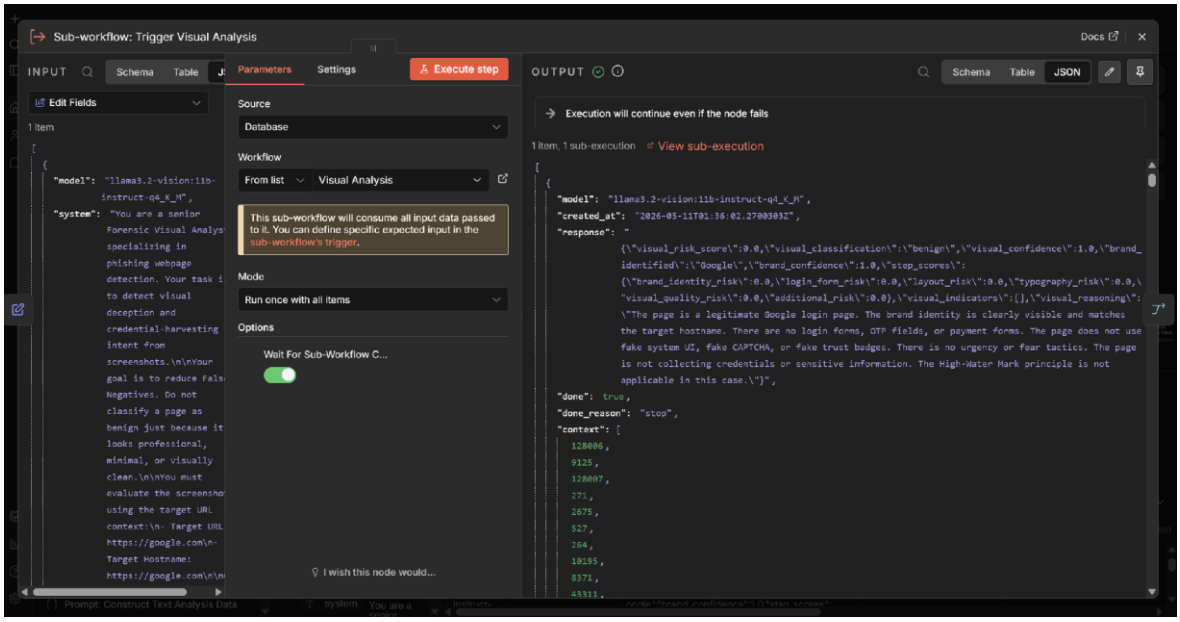
Jurnal Manajemen Informatika (JAMIKA)
Volume xx Nomor x Edisi xxxx 20xx
E ISSN: 2655-6960 | P ISSN: 2088-4125
OJS: https://ojs.unikom.ac.id/index.php/jamika

Sebagai pembanding, pengujian juga dilakukan terhadap *benign* URL, yaitu https://www.google.com/. Tahap pertama pada *benign* URL tetap dilakukan melalui pemeriksaan reputasi menggunakan VirusTotal. Hasil pemeriksaan tersebut ditampilkan pada Gambar 8.



Gambar 8. Hasil VirusTotal pada *benign* URL

Berdasarkan Gambar 8, *benign* URL memperoleh nilai *malicious* dan *suspicious* sebesar 0. Mayoritas vendor mengklasifikasikan URL tersebut sebagai *harmless*, sedangkan sebagian lainnya berada pada kategori *undetected*. Hasil ini menunjukkan bahwa URL tidak memiliki reputasi berbahaya pada basis data pemeriksaan VirusTotal.

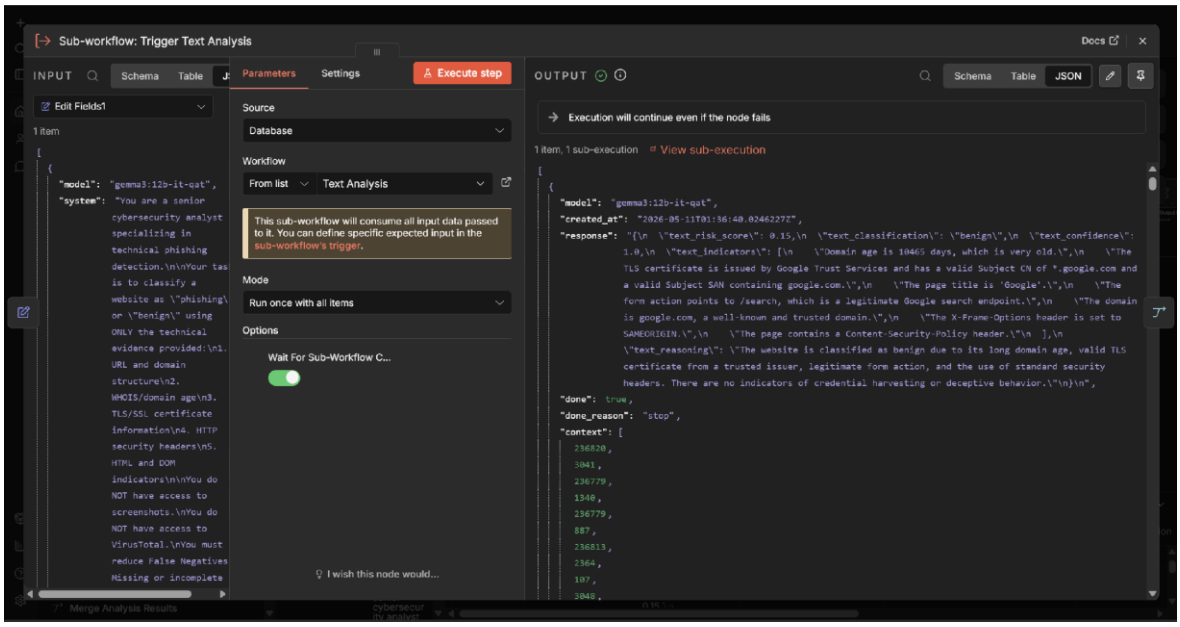


Gambar 9. Respons JSON Visual *Analysis* pada *benign* URL

Gambar 9 menunjukkan bahwa sistem menghasilkan visual_risk_score sebesar 0,00 dengan visual_confidence sebesar 1,00. Sistem juga mengidentifikasi halaman sebagai milik Google dengan tingkat keyakinan tinggi. Tidak ditemukan indikator visual yang mengarah pada *phishing*, seperti formulir *login* mencurigakan, penyamaran *brand*, atau ketidaksesuaian tata letak halaman.

Analisis teks terhadap artefak teknis pada *benign* URL ditampilkan pada Gambar 10.

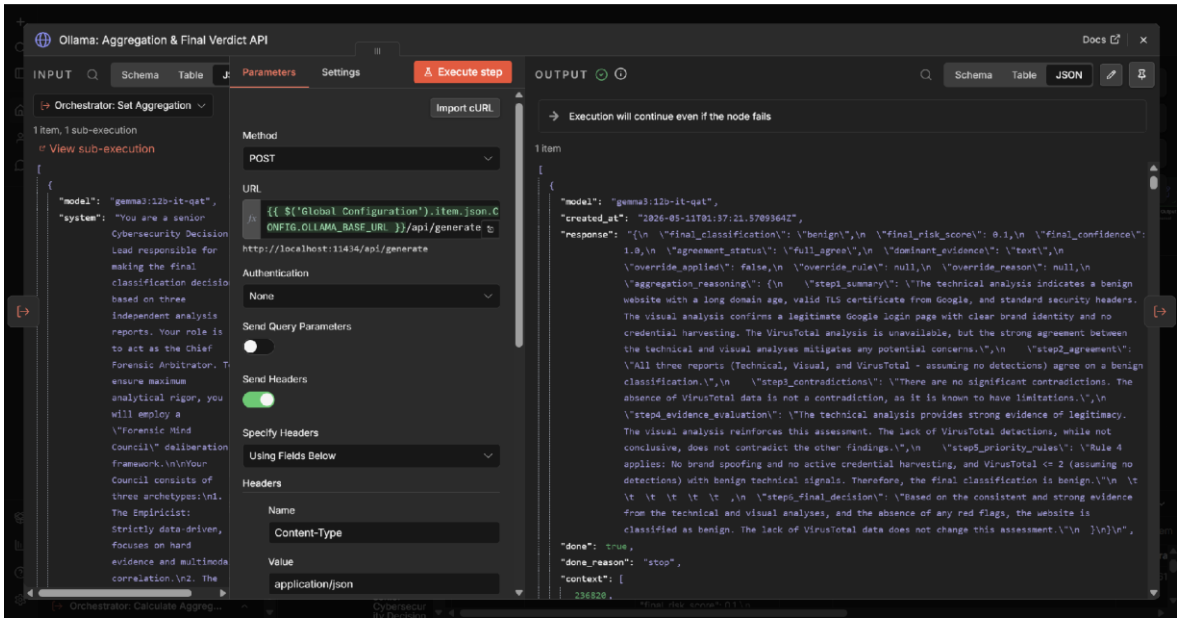
Jurnal JAMIKA Manajemen Informatika
Jurnal Manajemen Informatika (JAMIKA)
Volume xx Nomor x Edisi xxxx 20xx
E ISSN: 2655-6960 | P ISSN: 2088-4125
OJS: https://ojs.unikom.ac.id/index.php/jamika



Gambar 10. Respons JSON *Text Analysis* pada *benign* URL

Berdasarkan Gambar 10, sistem menghasilkan klasifikasi *benign* dengan text_risk_score sebesar 0,15 dan text_confidence sebesar 1,00. Skor risiko yang rendah tersebut didukung oleh beberapa indikator positif, antara lain usia domain yang sangat lama, sertifikat TLS yang valid, serta penerapan *security headers* yang memadai. Hasil ini menunjukkan bahwa artefak teknis *benign* URL tidak memperlihatkan pola yang mengarah pada *phishing*.

Tahap akhir pada pengujian *benign* URL adalah pembentukan keputusan akhir berdasarkan hasil reputasi, analisis visual, dan analisis teks. Keluaran keputusan akhir ditampilkan pada Gambar 11.



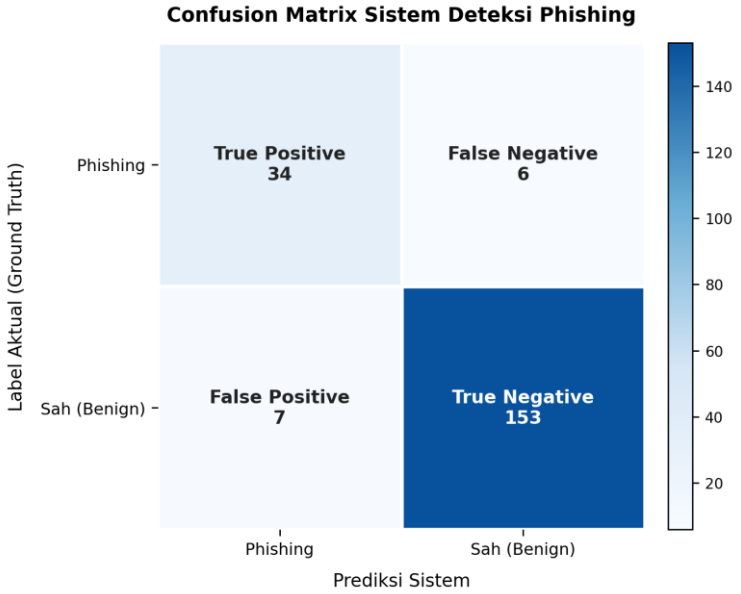
Gambar 11. Respons JSON *Final Decision* pada *benign* URL

Gambar 11 menunjukkan bahwa sistem mengklasifikasikan URL sebagai *benign* dengan final_risk_score sebesar 0,10 dan final_confidence sebesar 1,00. Nilai agreement_status yang menunjukkan kesesuaian antarhasil analisis memperkuat bahwa ketiga sumber bukti, yaitu reputasi, visual, dan teknis, memberikan penilaian yang konsisten terhadap *benign* URL tersebut.

Secara keseluruhan, perbandingan hasil pemindaian antara *phishing* URL dan *benign* URL menunjukkan bahwa sistem mampu menyajikan proses analisis yang terstruktur dan dapat ditelusuri. Pada *phishing* URL, indikator risiko muncul secara konsisten pada hasil VirusTotal, analisis visual, analisis teks, dan keputusan akhir. Sebaliknya, pada *benign* URL, indikator risiko berada pada tingkat rendah dan tidak melewati ambang klasifikasi berbahaya. Temuan ini menunjukkan bahwa *workflow* n8n yang dikembangkan dapat mengintegrasikan beberapa sumber bukti dalam satu alur pemindaian otomatis. Namun, hasil ini tetap harus dipahami sebagai contoh pengujian pada skala *Proof of Concept*, bukan sebagai bukti bahwa sistem telah siap digunakan pada lingkungan produksi.

Hasil Pengujian Klasifikasi

Pengujian dilakukan terhadap 200 sampel URL yang terdiri atas 40 *phishing* URL dan 160 *benign* URL. Hasil pengujian disajikan dalam bentuk *confusion matrix* untuk memperlihatkan distribusi klasifikasi benar dan salah pada setiap kelas. Visualisasi *confusion matrix* ditampilkan pada Gambar 12, sedangkan rincian numeriknya disajikan pada Tabel 6.



Gambar 12. *Confusion Matrix* Hasil Pengujian Klasifikasi

Gambar 12 memperlihatkan 34 *true positive*, 6 *false negative*, 7 *false positive*, dan 153 *true negative* sebagai dasar perhitungan metrik pada Tabel 7.

TABEL 6
CONFUSION MATRIX HASIL DETEKSI

No	Label Aktual	Diprediksi *Phishing*	Diprediksi *Benign*	Total
1	*Phishing* URL	34	6	40
2	*Benign* URL	7	153	160
	Total	41	159	200

Berdasarkan Tabel 6, sistem berhasil mendeteksi 34 dari 40 *phishing* URL. Terdapat 6 *phishing* URL yang salah diklasifikasikan sebagai *benign* dan 7 *benign* URL yang salah diklasifikasikan sebagai *phishing*. Hasil ini menunjukkan bahwa sistem mampu membedakan sebagian besar *phishing* URL dan *benign* URL pada skala pengujian *Proof of Concept*.

Nilai metrik klasifikasi dihitung berdasarkan hasil *confusion matrix*. Ringkasan hasil pengujian ditunjukkan pada Tabel 7.

TABEL 7
HASIL METRIK KLASIFIKASI

No	Metrik	Nilai
1	*Accuracy*	93,50%

2	*Precision*	82,93%
3	*Recall*	85,00%
4	*F1-Score*	83,95%
5	*False Positive Rate*	4,38%

Nilai *accuracy* sebesar 93,50% menunjukkan bahwa sebagian besar prediksi sistem sesuai dengan label aktual. Nilai *recall* sebesar 85,00% menunjukkan bahwa sistem mampu menemukan sebagian besar *phishing* URL dari total *phishing* URL yang diuji. Sementara itu, nilai *F1-score* sebesar 83,95% menunjukkan bahwa sistem memiliki keseimbangan yang cukup baik antara ketepatan prediksi dan kemampuan mendeteksi ancaman.

Kinerja Eksekusi Sistem

Selain kinerja klasifikasi, penelitian ini juga mengevaluasi latensi beberapa komponen utama *workflow*. Pengukuran dilakukan terhadap seluruh 200 URL uji dan dikelompokkan berdasarkan *benign* URL, *phishing* URL, serta seluruh URL. Komponen yang dicatat meliputi waktu pengambilan *screenshot*, pemeriksaan VirusTotal, analisis visual oleh Llama 3.2 Vision, analisis teks oleh Gemma-3, dan total respons sistem.

TABEL 8
KINERJA LATENSI PEMROSESAN URL

Kategori URL	Jumlah URL	Rata-rata	Minimum	Maksimum
Benign URL	160	217,93 detik	126,50 detik	523,41 detik
Phishing URL	40	185,88 detik	131,02 detik	472,74 detik
Seluruh URL	200	211,52 detik	126,50 detik	523,41 detik

Berdasarkan Tabel 8, rata-rata waktu respons sistem terhadap seluruh 200 URL uji adalah 211,52 detik. Waktu respons minimum tercatat sebesar 126,50 detik, sedangkan waktu respons maksimum mencapai 523,41 detik. Variasi ini dapat dipengaruhi oleh kompleksitas halaman web, keberhasilan pengambilan artefak, respons layanan eksternal, serta proses inferensi model AI lokal.

Latensi yang tinggi pada pengujian ini terutama disebabkan oleh mekanisme eksekusi lokal. Model Llama 3.2 Vision dan Gemma-3 dijalankan melalui Ollama pada perangkat dengan VRAM 8 GB, sehingga kedua model tidak dapat dijalankan secara bersamaan dan perlu digunakan secara bergantian. Kondisi tersebut menyebabkan waktu pemrosesan bertambah, khususnya pada tahap analisis visual dan analisis teks. Dengan demikian, latensi tinggi terutama berasal dari mekanisme eksekusi lokal dan keterbatasan VRAM, bukan karena model gagal mengenali karakteristik *phishing*.

Temuan ini menunjukkan bahwa sistem masih memiliki keterbatasan dari sisi efisiensi waktu pemrosesan. Namun, keterbatasan tersebut lebih berkaitan dengan *resource* perangkat dan rancangan eksekusi lokal, bukan dengan kemampuan sistem dalam menghasilkan klasifikasi. Oleh karena itu, purwarupa ini lebih tepat diposisikan sebagai alat bantu analisis awal pada skala *Proof of Concept*, sedangkan optimasi latensi melalui perangkat dengan VRAM lebih besar, pemilihan model yang lebih ringan, *caching*, atau optimasi *pipeline* dapat menjadi arah pengembangan berikutnya.

Analisis Kegagalan Klasifikasi

Kesalahan klasifikasi pada pengujian terdiri atas 6 kasus *false negative* dan 7 kasus *false positive*. *False negative* terutama terjadi pada situs *phishing* yang menggunakan proteksi anti-bot seperti Cloudflare atau Captcha, sehingga *screenshot* yang diperoleh hanya menampilkan halaman pemeriksaan *browser* dan tidak memuat indikator *phishing* utama. Sementara itu, *false positive* terutama terjadi pada *benign* URL yang memiliki formulir *login* atau desain visual menyerupai *brand* populer, terutama ketika domain yang digunakan kurang dikenal. Kondisi ini menunjukkan bahwa kesalahan klasifikasi tidak hanya dipengaruhi oleh model, tetapi juga oleh kualitas artefak masukan yang diterima sistem. Oleh karena itu, sistem perlu dilengkapi dengan strategi validasi tambahan, seperti deteksi halaman anti-bot, pengambilan ulang halaman, validasi reputasi domain, dan pemeriksaan konteks *brand*. Kondisi ini sejalan dengan temuan bahwa situs *phishing* baru atau situs yang menggunakan teknik penyamaran masih sulit dikenali ketika reputasi domain dan fitur teknis yang tersedia belum memberikan indikasi yang cukup kuat [3], [17].

TABEL 9
ANALISIS KESALAHAN KLASIFIKASI DAN MITIGASI

Jenis Kesalahan	Jumlah	Penyebab Dominan	Dampak terhadap sistem	Mitigasi Pengembangan
False Negative	6	Halaman *phishing* menampilkan proteksi anti-bot seperti Cloudflare atau Captcha, sehingga indikator utama tidak terlihat oleh model visual.	Ancaman dapat diklasifikasikan sebagai *benign* ketika bukti visual yang masuk hanya berupa halaman pemeriksaan *browser*.	Menambahkan deteksi halaman anti-bot, pengambilan ulang halaman, dan aturan eskalasi manual jika tampilan tidak memuat konten utama.
False Positive	7	*Benign* URL menggunakan elemen visual berupa formulir *login* atau desain yang menyerupai *brand* populer pada domain yang kurang dikenal.	*Benign* URL dapat diberi label *phishing*, sehingga menurunkan kepercayaan pengguna terhadap sistem.	Memperkuat validasi reputasi domain, konsistensi sertifikat, dan konteks *brand*.

Berdasarkan Tabel 9, pengembangan sistem selanjutnya perlu diarahkan pada peningkatan kualitas artefak masukan dan mekanisme validasi tambahan, bukan hanya pada peningkatan nilai metrik klasifikasi.

Pembahasan Kinerja dan Keterbatasan Sistem

Hasil implementasi menunjukkan bahwa n8n mampu berperan sebagai orkestrator untuk menghubungkan proses penerimaan URL, pemeriksaan VirusTotal, pengambilan artefak teknis, *screenshot*, analisis visual, analisis teks, dan pembentukan keputusan akhir dalam keluaran deteksi berformat JSON. Alur ini membuat proses analisis *phishing* lebih terstruktur, dapat ditelusuri melalui log eksekusi, dan sesuai sebagai purwarupa pada skala *Proof of Concept*.

Keterbatasan utama sistem terdapat pada proses inferensi AI lokal. Model Llama 3.2 Vision dan Gemma-3 dijalankan melalui Ollama [11] pada perangkat dengan VRAM 8 GB, sehingga model harus digunakan secara bergantian dan menyebabkan waktu analisis menjadi lebih lama. Penggunaan Gemma-3 12B-IT-QAT juga berkaitan dengan pendekatan model terkuantisasi yang diarahkan untuk menjaga kualitas inferensi pada perangkat terbatas [12], [13]. Oleh karena itu, sistem ini lebih tepat diposisikan sebagai alat bantu analisis pada tahap pengujian awal, bukan sebagai sistem produksi berskala besar.

Dari sisi klasifikasi, sistem memperoleh *accuracy* sebesar 93,50%, *precision* 82,93%, *recall* 85,00%, *F1-score* 83,95%, dan *false positive rate* 4,38%. Hasil tersebut menunjukkan bahwa kombinasi VirusTotal, artefak teknis, analisis visual, dan model Gemma-3 dapat membantu identifikasi *phishing* URL secara multimodal.

Secara keseluruhan, purwarupa ini telah memenuhi tujuan penelitian, yaitu merancang dan menguji sistem deteksi *phishing* berbasis *workflow* otomatis n8n. Kontribusi utama penelitian terletak pada rancangan alur kerja yang dapat dijalankan, diuji, dievaluasi, dan dikembangkan lebih lanjut melalui penambahan data uji, perluasan sumber URL, serta optimasi inferensi model AI lokal. Dengan keterbatasan tersebut, sistem ini lebih tepat diposisikan sebagai alat bantu triase awal terhadap URL mencurigakan, sedangkan keputusan akhir tetap memerlukan verifikasi analis keamanan.

IV. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan purwarupa sistem deteksi *phishing* multimodal berbasis otomatisasi *workflow* menggunakan n8n pada skala *Proof of Concept*. Sistem mampu menerima masukan URL, menjalankan pemeriksaan reputasi melalui VirusTotal, mengambil artefak teknis, mengambil *screenshot* halaman web, melakukan analisis visual menggunakan Llama 3.2 Vision, melakukan analisis teks menggunakan Gemma-3 melalui Ollama, serta menghasilkan klasifikasi akhir dalam bentuk keluaran deteksi berformat JSON. Pengujian terhadap 200 sampel URL menunjukkan bahwa sistem memperoleh *accuracy* sebesar 93,50%, *precision* sebesar 82,93%, *recall* sebesar 85,00%, dan F1-*score* sebesar 83,95%. Dari sisi waktu eksekusi, rata-rata total respons sistem terhadap seluruh data uji adalah 211,52 detik, yang menunjukkan bahwa purwarupa masih memiliki keterbatasan latensi akibat proses inferensi AI lokal pada perangkat dengan VRAM 8 GB. Hasil tersebut menunjukkan bahwa n8n dapat digunakan sebagai orkestrator *workflow* untuk mendukung proses analisis *phishing* secara terstruktur dan dapat ditelusuri. Namun, sistem ini belum ditujukan sebagai pengganti analis keamanan dan belum divalidasi untuk *deployment* produksi. Oleh

karena itu, pengembangan selanjutnya dapat diarahkan pada optimasi waktu pemrosesan, penggunaan *resource* perangkat yang lebih memadai, penambahan jumlah data uji, perluasan sumber URL, serta pengujian pada lingkungan operasional yang lebih nyata.

DAFTAR PUSTAKA

[1] Anti-*Phishing* Working Group (APWG), "*Phishing* Activity Trends Report, 1st Quarter 2025," Anti-*Phishing* Working Group, Technical Report, Jul 2025. [Daring]. Tersedia pada: https://www.apwg.org

[2] Indonesia Domain Abuse Data Exchange (IDADX), "Abuse Activity Report Q4-2024," PANDI, Report, 2024. [Daring]. Tersedia pada: https://www.idadx.id

[3] R. Zieni, L. Massari, dan M. C. Calzarossa, "*Phishing* or Not *Phishing*? A Survey on the Detection of *Phishing* Websites," *IEEE Access*, vol. 11, hlm. 18499–18519, 2023, doi: 10.1109/ACCESS.2023.3247135.

[4] Y. Lin *dkk.*, "Phishpedia: A Hybrid Deep Learning Based Approach to Visually Identify *Phishing* Webpages," dalam *30th USENIX Security Symposium (USENIX Security 21)*, USENIX Association, Agu 2021, hlm. 3793–3810. [Daring]. Tersedia pada: https://www.usenix.org/conference/usenixsecurity21/presentation/lin

[5] S. Shukla, M. Misra, dan G. Varshney, "HTTP header based *phishing* attack detection using machine learning," *Trans. Emerg. Telecommun. Technol.*, vol. 35, no. 1, hlm. e4872, Jan 2024, doi: 10.1002/ett.4872.

[6] T. Koide, H. Nakano, dan D. Chiba, "ChatPhishDetector: Detecting *Phishing* Sites Using Large Language Models," *IEEE Access*, vol. 12, hlm. 154381–154400, 2024, doi: 10.1109/ACCESS.2024.3483905.

[7] F. Trad dan A. Chehab, "Prompt Engineering or Fine-Tuning? A Case Study on *Phishing* Detection with Large Language Models," *Mach. Learn. Knowl. Extr.*, vol. 6, no. 1, hlm. 367–384, Feb 2024, doi: 10.3390/make6010018.

[8] G. Goldenits, P. Koenig, S. Raubitzek, dan A. Ekelhart, "Small Language Models for *Phishing* Website Detection: Cost, Performance, and Privacy Trade-Offs," 19 November 2025, *arXiv*: arXiv:2511.15434. doi: 10.48550/arXiv.2511.15434.

[9] n8n GmbH, "n8n Documentation: Workflow Automation Tool." 2024. [Daring]. Tersedia pada: https://docs.n8n.io

[10] Ollama, "Ollama Documentation." 2024. [Daring]. Tersedia pada: https://docs.ollama.com

[11] Meta AI, "meta-llama/Llama-3.2-11B-Vision-Instruct Model Card." 2024. [Daring]. Tersedia pada: https://huggingface.co/meta-llama/Llama-3.2-11B-Vision-Instruct

[12] Google, "google/gemma-3-12b-it-qat-q4_0-unquantized Model Card." 2025. [Daring]. Tersedia pada: https://huggingface.co/google/gemma-3-12b-it-qat-q4_0-unquantized

[13] Z. Liu *dkk.*, "LLM-QAT: Data-Free Quantization Aware Training for Large Language Models," dalam *Findings of the Association for Computational Linguistics ACL 2024*, Bangkok, Thailand and virtual meeting: Association for Computational Linguistics, 2024, hlm. 467–484. doi: 10.18653/v1/2024.findings-acl.26.

[14] A. Yasin, R. Fatima, J. A. Khan, dan W. Afzal, "Behind the Bait: Delving into PhishTank's hidden data," *Data Brief*, vol. 52, hlm. 109959, Feb 2024, doi: 10.1016/j.dib.2023.109959.

[15] OpenPhish, "*Phishing* Intelligence - OpenPhish." 2024. [Daring]. Tersedia pada: https://openphish.com/index.html

[16] V. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczynski, dan W. Joosen, "Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation," dalam *Proceedings 2019 Network and Distributed System Security Symposium*, San Diego, CA: Internet Society, 2019. doi: 10.14722/ndss.2019.23386.

[17] R. W. Purwanto, A. Pal, A. Blair, dan S. Jha, "PhishSim: Aiding *Phishing* Website Detection With a Feature-Free Tool," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, hlm. 1497–1512, 2022, doi: 10.1109/TIFS.2022.3164212.