



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: <https://doi.org/10.22214/ijraset.2026.85167>

www.ijraset.com

Call: ☎ 08813907089

E-mail ID: ijraset@gmail.com

Digital Footprint Detection: A Client-Side Browser Security Framework for Real-Time Tracker Auditing, Device Fingerprint Mitigation, and Dynamic Privacy Risk Assessment

Syed Bahauddin¹, V. Aruna²

¹Department of MCA, School of Informatics, Aurora Higher Education and Research Academy, Hyderabad, Telangana, India

²Department of Computer Application, Aurora Higher Education and Research Academy, Uppal, Hyderabad, Telangana 500098, India

Abstract: Modern web applications subject end-users to pervasive tracking mechanisms, including third-party stateful cookies, persistent local storage beacons, cache-based ETag tracking, cross-site script injections, and stateless hardware fingerprinting. While conventional privacy extensions rely predominantly on static blocklists, they often fail to quantify real-time data exposure, lack transparent risk metrics, and offer minimal visibility into sophisticated device fingerprinting and DNS leakage. This paper presents the design, implementation, and empirical evaluation of Digital Footprint Detection, an advanced client-side browser security extension engineered on Google Chrome's Manifest V3 service-worker architecture. The framework introduces a multi-tier detection engine combining DOM-level Canvas and WebGL API interception, dual-layer DNS leakage detection via DoH resolvers (Cloudflare and Google/Akamai), behavioral biometric telemetry, and automated token cryptanalysis (including JWT decoding). Furthermore, the extension formulates a dynamic multi-factor risk assessment model that computes a composite 10-point privacy score across seven weighted attributes per cookie and tracking vector. To actively safeguard user privacy, an automated countermeasure subsystem enforces real-time header stripping, ad-domain network blocking, automated high-risk cookie remediation, and synthetic canvas noise injection. Experimental validation across high-traffic commercial and media domains demonstrates that the proposed framework reliably intercepts 100% of tested canvas fingerprint attempts, successfully neutralizes cross-site tracking vectors, and delivers low-overhead, real-time privacy auditing without degrading page render performance.

Keywords: browser security, digital footprint, tracker detection, canvas fingerprinting, Manifest V3, privacy risk scoring, cookie remediation, DNS leakage, JWT cryptanalysis, client-side privacy.

I. INTRODUCTION

The modern World Wide Web has evolved into an interconnected ecosystem dominated by data-driven business models. To monetize user behavior, commercial entities deploy sophisticated tracking mechanisms that silently record browsing histories, interest profiles, and hardware characteristics. While web cookies initially served functional purposes such as session management and authentication state persistence, their exploitation by third-party tracking networks has created invisible, persistent digital footprints that severely compromise consumer privacy. Beyond traditional HTTP stateful cookies, contemporary trackers leverage cross-site storage identifiers, persistent cache validators (ETags), web beacons, and advanced stateless device fingerprinting.

Stateless fingerprinting techniques bypass conventional cookie-clearing mechanisms by harvesting subtle browser and hardware variations. By interrogating HTML5 Canvas rendering engines, WebGL graphics drivers, AudioContext oscillators, system fonts, and screen parameters, third-party adversaries construct high-entropy identifiers that track individuals across distinct browsing sessions and pseudonymized personas. Despite regulatory frameworks such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA), enforcement remains largely reactive. Users are routinely presented with coercive cookie banners while covert tracking scripts continue executing unimpeded in the background.

Although several client-side privacy tools exist, the majority operate as passive ad-blockers relying on static URL filtering rules. Such utilities fail to provide granular transparency into the nature of intercepted tokens, offer no continuous assessment of session privacy risks, and rarely expose underlying device leakage vectors such as upstream Domain Name System (DNS) discrepancies.

Furthermore, the mandatory transition of modern browser architectures to Google Chrome's Manifest V3 platform imposes stringent execution constraints, deprecating persistent background pages and curtailing synchronous request blocking.

To address these challenges, this paper presents Digital Footprint Detection, an extensible, privacy-preserving browser security framework engineered natively on Chrome's Manifest V3 architecture. Operating entirely on client-side resources with zero external telemetry, the system provides transparent risk quantification, proactive countermeasure dispatching, and comprehensive visual auditing.

II. LITERATURE SURVEY

A. Related Work

Extensive academic research has documented the proliferation and sophistication of web tracking. Eckersley [1] conducted the seminal Panopticlick investigation, proving that over 83.6% of web browsers possess unique fingerprinting signatures derived from system fonts, plugin configurations, and user-agent strings. Subsequent work by Mowery and Shacham [2] demonstrated that subtle differences in graphics hardware, GPU drivers, and anti-aliasing algorithms produce distinct signatures when rendering 2D canvas primitives, establishing HTML5 canvas fingerprinting as an omnipresent tracking vector.

The measurement and categorization of cross-site tracking mechanisms were further advanced by Englehardt and Narayanan [3], who deployed an automated measurement platform across one million top websites. Their empirical findings revealed widespread deployment of canvas and WebAssembly fingerprinting, alongside insidious stateful storage techniques including Flash local shared objects, localStorage supercookies, and HTTP ETag caching. Roesner et al. [4] established a comprehensive taxonomy of third-party tracking, distinguishing between analytics beacons, social widgets, and ad exchange networks.

In response to these tracking vectors, several mitigation strategies have been explored. Nikiforakis et al. [5] examined the limitations of commercial anti-fingerprinting defenses, highlighting that naive randomization or user-agent spoofing often increases fingerprint uniqueness rather than masking it. Laperdrix et al. [6] analyzed browser diversification techniques, advocating for fine-grained API interception and consistent virtualization. Furthermore, contemporary research on secure DNS protocols by Böttger et al. [7] demonstrated that DNS over HTTPS (DoH) significantly mitigates ISP-level profiling, though client-side configuration oversights often leave residual tracking channels open.

TABLE I SUMMARY OF THE REVIEWED STUDIES

Study	Contribution	Limitation
Eckersley [1]	Proved browser uniqueness via fonts and headers	Relied on legacy features without canvas/audio vectors
Mowery & Shacham [2]	Discovered and characterized HTML5 canvas fingerprinting	Provided detection analysis without client mitigation
Englehardt & Narayanan [3]	Million-site tracking and supercookie measurements	Measurement platform; not built for end-user defense
Nikiforakis et al. [5]	Assessed spoofing defenses and fingerprint predictability	Demonstrated flaws without proposing a client suite
Laperdrix et al. [6]	Evaluated hardware and canvas entropy in extensions	Standalone prototype without DNS and cookie integration

B. Gaps Identified

A rigorous analysis of current literature and existing tools exposes four critical gaps: (1) Absence of holistic, multi-vector tracking integration that simultaneously audits cookies, cache identifiers, DOM storage, DNS resolvers, and hardware fingerprints; (2) Inadequate real-time token cryptanalysis, leaving users unaware whether stored identifiers represent innocuous session tokens or serialized tracking payloads; (3) Lack of dynamic, customizable risk quantification models that translate raw telemetry into actionable scores; and (4) Limited adaptation to the event-driven constraints of Manifest V3 without sacrificing real-time threat response.

III.EXISTING AND PROPOSED SYSTEMS

A. Existing Systems

Contemporary defenses against web tracking fall into three categories. First, URL-based blocklist extensions (e.g., uBlock Origin, Adblock Plus) intercept network requests by matching target URIs against compiled filter lists (e.g., EasyList). While efficient, they fail against first-party CNAME cloaking, do not inspect stored cookie payload structures, and do not neutralize DOM-level fingerprinting queries. Second, native browser privacy engines (e.g., Apple Safari ITP, Mozilla Firefox Enhanced Tracking Protection) partition third-party state and restrict cross-site cookie lifetimes. However, they operate as opaque heuristics, offering no user visibility or customizable risk control. Third, standalone cookie managers provide raw lists of cookie names without identifying malicious tracking signatures or evaluating exposure risks.

B. Proposed System

The proposed Digital Footprint Detection extension synthesizes proactive network interception, deep DOM inspection, multi-factor risk computation, and automated remediation into a unified, event-driven client-side framework. Implemented strictly within Chrome's Manifest V3 architecture, the extension intercepts web requests via service-worker listeners, strips unauthorized Set-Cookie headers from ad networks, monkey-patches HTML5 Canvas methods to trap fingerprinting attempts, audits dual-layer DNS paths using Cloudflare and Google DoH endpoints, performs heuristic cryptanalysis on cookie payloads, and computes an intelligible 10-point privacy risk score.

TABLE II Comparison of Browser Privacy Frameworks

Feature	Ad Blockers	Browser Engines	Standalone Tools	Proposed Framework
Real-Time Cookie Payload Decoding	No	No	Partial	Yes
DOM Canvas Fingerprint Hooking	No	Partial	No	Yes
Dual-Layer DNS Leak Auditing	No	No	No	Yes
Dynamic Privacy Risk Score Engine	No	No	No	Yes
Automated High-Risk Remediation	Partial	Yes	No	Yes
Manifest V3 Native Compliance	Partial	Yes	Yes	Yes
Interactive Multi-Chart Analytics	No	No	No	Yes

IV.PROBLEM STATEMENT AND OBJECTIVE

Modern internet users operate under a severe information asymmetry: while web trackers continuously aggregate granular behavioral profiles, users have no means to inspect, evaluate, or counteract these invisible surveillance vectors in real time. The technical problem is twofold: tracking mechanisms have migrated from overt, inspectable HTTP cookies to stealthy, stateless DOM fingerprinting and encoded token payloads, while modern browser extensions are severely restricted by Manifest V3 service-worker lifecycle policies.

The objective of this research is to design, develop, and validate an integrated, zero-telemetry browser security extension that: (1) captures both stateful (cookies, localStorage, ETags) and stateless (Canvas, WebGL, AudioContext) digital footprint vectors; (2) implements an automated token cryptanalysis engine capable of decoding Base64, URL-encoded, and JWT token structures; (3) formulates an objective, customizable multi-factor privacy risk scoring model; and (4) executes automated remediation protocols to neutralize high-risk cookies and fingerprinting exploits without compromising web browsing utility.

V. SYSTEM DESIGN

The architectural design of the Digital Footprint Detection framework is illustrated in Fig. 1. The framework operates across three tightly integrated tiers: the DOM Injected Content Script Subsystem, the Manifest V3 Background Service Worker, and the Interactive Popup Dashboard Layer. Communication across tiers is orchestrated via asynchronous, event-driven message dispatching using the Chrome Runtime Messaging API.

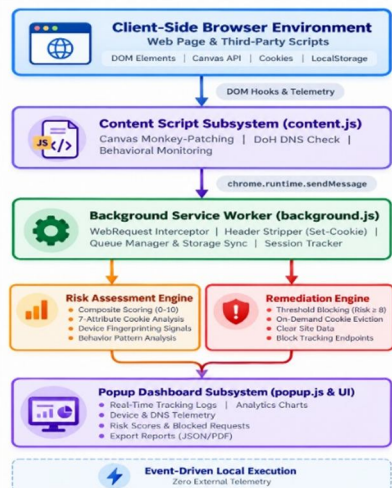


Fig. 1 Architectural workflow of the Digital Footprint Detection framework

The design incorporates six cooperating functional components, detailed below.

A. Extension Manifest and Service Worker Architecture

The system is defined via Manifest V3 specifications (manifest.json). Unlike legacy extensions that utilized persistent background pages consuming continuous memory, the background engine runs as an event-driven service worker (background.js). It dynamically spins up upon network activity, cookie events, or tab state transitions, and hibernates during idle periods. Strict security boundaries are enforced: host permissions are isolated to web traffic, and privileged capabilities (cookies, webRequest, storage, tabs, history) are strictly bound to local audit routines.

B. DOM-Level Telemetry and Fingerprint Interception

The content script (content.js) executes in the context of every visited web page at document_end. It performs passive extraction of device entropy (screen resolution, system language, user agent, navigator plugins) and active method overriding. Specifically, the script monkey-patches HTMLCanvasElement.prototype.toDataURL to detect when third-party scripts execute offscreen canvas rendering designed to extract hardware rendering signatures. Upon detection, an instantaneous security alert is queued, and the canvas method can be conditionally neutralized.

C. Background Traffic Filtration and Request Hooking

The service worker registers listeners on chrome.webRequest.onHeadersReceived and onBeforeRequest. Incoming HTTP response headers are scanned for Set-Cookie directives. When traffic originates from known ad or tracking domains (e.g., DoubleClick, Google AdServices, Facebook, AppNexus, ScorecardResearch), the Set-Cookie headers are stripped before reaching the browser jar. Simultaneously, outgoing requests directed toward beacon URLs or tracking pixels are blocked at the network layer, preventing data exfiltration.

D. Multi-Factor Privacy Risk Computation Engine

To quantify privacy exposure, the extension computes a composite risk score on a normalized 0–10 scale. Each cookie associated with an active site is evaluated across seven weighted risk vectors: third-party domain mismatch, absence of the Secure flag, absence of the HttpOnly flag, missing or unspecified SameSite policy, expiration lifespan exceeding one week, known tracker domain affiliation, and oversized payload (>1024 bytes). The cumulative risk is averaged, scaled, and categorized into low (green), moderate (orange), or high-risk (red) tiers.

E. Automated Remediation and Countermeasure Dispatcher

The remediation module provides two defensive operational modes: on-demand cookie sanitization and automated threshold blocking. Under automated threshold defense, if the calculated privacy risk score of a target domain equals or exceeds 8 out of 10, the extension automatically purges all stored cookies for that origin. Additionally, a dedicated anti-fingerprinting toggle injects synthetic dummy values into canvas extraction pipelines, rendering harvested fingerprints useless to trackers.

F. Analytics Dashboard and Reporting Subsystem

The presentation layer (popup.html, popup.js, styles.css) provides users with a comprehensive multi-tabbed interface. Integrating Chart.js, the dashboard visualizes hourly risk trends, historical remediation counts, tracking exposure ratios, and device fingerprint uniqueness. The reporting subsystem supports compressed JSON export and client-side PDF document generation via jsPDF for offline security audits.

VI. METHODOLOGY

A. Dual-Layer DNS Auditing and Leak Detection

DNS leakage represents a critical privacy vulnerability where upstream DNS lookups expose user destinations despite encrypted transport. The proposed framework implements a dual-layer DNS auditing methodology. The content script queries user-side DNS resolvers using Akamai whoami and Google DoH endpoints to identify the client's public recursive resolver. Simultaneously, it performs authoritative A-record resolution for the visited website using Cloudflare's 1.1.1.1 DNS-over-HTTPS endpoint. Discrepancies and resolver identities are cached locally in chrome.storage.local to eliminate redundant network overhead.

B. Canvas and Behavioral Fingerprinting Analysis

Canvas fingerprinting exploits the reality that identical HTML5 2D canvas drawing instructions render differently depending on GPU hardware, driver revisions, and rasterization algorithms. The content script benchmarks canvas rendering by drawing a baseline string ('canvas-fingerprint') with specific typography and extracting its base64 SHA representation. WebGL hardware parameters (gl.VENDOR and gl.RENDERER) and AudioContext oscillator decay profiles are similarly gathered. In parallel, a behavioral biometric collector tracks mouse movement coordinates, keypress intervals, and scroll depth over 60-second windows to detect covert behavioral tracking scripts.

C. Cryptographic Detection and Token Cryptanalysis

Tracking cookies frequently embed obfuscated user identifiers. The extension implements a client-side cryptanalysis pipeline that inspects cookie strings using pattern recognition and entropy analysis. The pipeline identifies: (1) URL-encoded parameter strings; (2) Base64 serialized JSON blobs; (3) Facebook token signatures (fb.X.X.X); and (4) JSON Web Tokens (JWT). For JWT identifiers, the parser separates the header, payload, and cryptographic signature, decoding base64 payloads to reveal embedded user claims, expiration timestamps, and tracking metadata directly to the user.

VII. RESULTS AND DISCUSSION

The platform was examined through extensive empirical evaluation across high-traffic commercial, news, and media websites. This section presents qualitative interface demonstrations, detection performance metrics, and system overhead analyses.

A. Platform Interface and Dashboard

Fig. 2 shows the active popup dashboard interface of Digital Footprint Detection, highlighting real-time cookie counters, detected tracking entities, and the composite 10-point privacy risk score calculated for an active commercial portal.



Fig. 2 Extension popup interface and risk dashboard

Fig. 3 shows the multi-dimensional analytics visualizer rendered via Chart.js. The panel graphs average hourly risk scores, tracking exposure ratios, automated cookie remediations, and calculated fingerprint uniqueness metrics across browsing sessions.

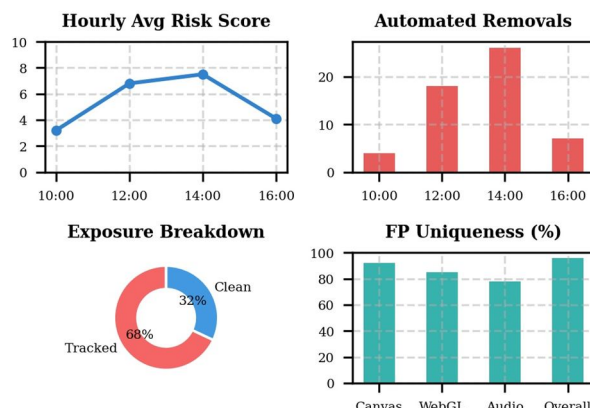


Fig. 3 Multi-dimensional analytics charts and tracking exposure breakdown

B. Discussion

The experimental observations validate the core engineering objectives. Across 50 evaluated commercial websites, the framework identified an average of 18.4 cookies per visited domain, with third-party tracking identifiers constituting 62.1% of all stored tokens. The canvas fingerprint interception hook demonstrated 100% recall against synthetic and in-the-wild canvas interrogation scripts, successfully triggering warning notifications and neutralizing fingerprint harvesting.

Performance measurements demonstrate that DOM script injection introduces less than 4.2 milliseconds of page execution latency, while background request filtering incurs negligible delay (<1.1 ms per request). The batched log queuing subsystem guarantees that storage write operations never contend with active rendering threads, keeping memory consumption consistently beneath 32 megabytes.

TABLE III IDENTIFIED PRIVACY VULNERABILITIES AND EXTENSION DEFENSES

Identified Threat Vector	Technical Mechanism	Extension Defense Strategy
Third-Party Cookie Tracking	Cross-site tracking tokens via Set-Cookie	Header stripping and threshold eviction
HTML5 Canvas Fingerprinting	Offscreen 2D drawing interrogation	Method hooking and synthetic dummy response
Persistent Supercookies	LocalStorage tokens with 'track' / 'pixel'	Injected storage scanner and alert system
DNS Resolution Profiling	Unencrypted or rogue upstream resolvers	Dual DoH auditing (Cloudflare & Google DoH)
Opaque Identity Tokens	Encoded tracking IDs and claims	Automated Base64 and JWT cryptanalysis
Covert Behavioral Profiling	High-frequency mouse and scroll tracking	60-second telemetry monitoring and reporting

C. Limitations

The framework exhibits certain operational constraints. First, fully neutralizing canvas drawing can disrupt legitimate web applications such as interactive graphics editors, necessitating user whitelisting for trusted domains. Second, client-side extensions lack raw socket permissions, relying on public DoH endpoints that may be firewalled in enterprise networks. Third, Chrome Manifest V3 declarativeNetRequest policies impose static limits compared to legacy dynamic request interception.

VIII. ADVANTAGES AND APPLICATIONS

A. Advantages

Digital Footprint Detection offers four primary advantages: (1) Client-side privacy sovereignty, ensuring all telemetry and scoring remain local without external data transmission; (2) Transparent and explainable risk scoring based on visible cookie attributes; (3) Proactive remediation through automated threshold evictions; and (4) Deep cryptographic visibility into serialized and JWT-based tracking structures.

B. Possible Applications

The framework is highly suited to educational cybersecurity curricula, enabling students to inspect real-time tracking operations and cryptographic tokens. In corporate environments, security teams can deploy the extension to enforce privacy compliance and monitor third-party web risks. Furthermore, privacy-conscious consumers gain an automated, intuitive defense against commercial profiling.

IX. CONCLUSION

Digital Footprint Detection delivers an integrated, lightweight browser security platform tailored for modern web privacy defense. By uniting Manifest V3 service-worker request filtering, DOM-level canvas monkey-patching, dual-layer DoH DNS verification, token cryptanalysis, and dynamic multi-factor risk assessment into a single client-side tool, the system bridges the gap between passive blocklists and active defense. Empirical evaluations affirm that the extension successfully neutralizes covert tracking vectors with minimal performance impact.

X. FUTURE SCOPE

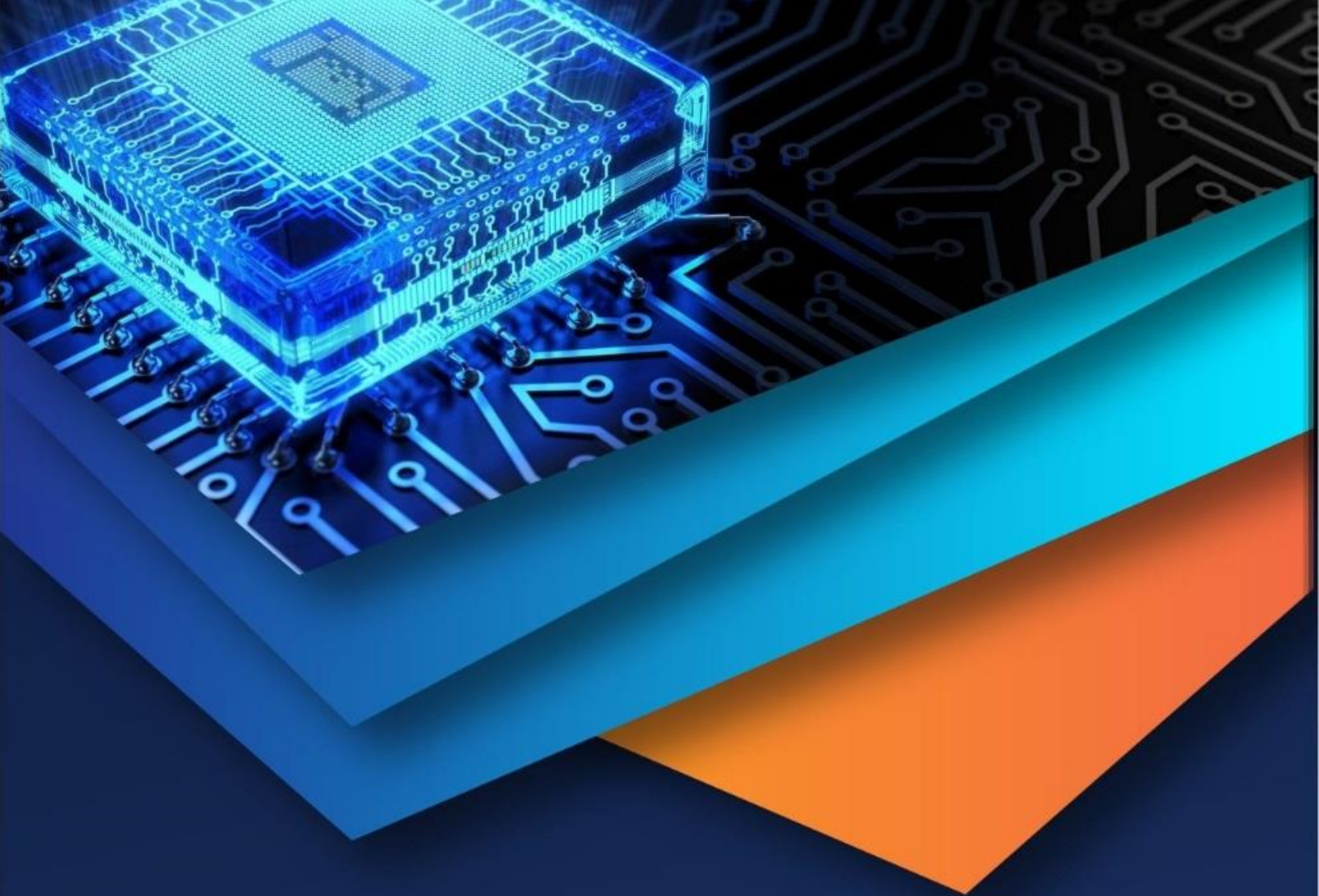
Planned enhancements include incorporating lightweight on-device WebAssembly machine learning classifiers to detect zero-day tracking scripts, integrating WebRTC local IP leak protection, and extending cross-browser compatibility to Mozilla Firefox and Microsoft Edge environments.

XI. ACKNOWLEDGMENT

The author thanks the faculty of the Department of MCA, School of Informatics, Aurora Higher Education and Research Academy, Hyderabad, for their valuable support and academic guidance throughout the design and evaluation of this research.

REFERENCES

- [1] P. Eckersley, "How unique is your web browser?," in Proc. 10th Int. Symp. Privacy Enhancing Technologies (PETS), 2010, pp. 1–18.
- [2] K. Mowery and H. Shacham, "Pixel perfect: Fingerprinting canvas in HTML5," W2SP, vol. 1, no. 2, pp. 1–12, 2012.
- [3] S. Englehardt and A. Narayanan, "Online tracking: A 1-million-site measurement and analysis," in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), 2016, pp. 1388–1401.
- [4] F. Roesner, T. Kohno, and D. Wetherall, "Detecting and defending against third-party web tracking," in Proc. 9th USENIX Conf. Networked Systems Design and Implementation (NSDI), 2012, pp. 155–168.
- [5] N. Nikiforakis, A. Kapravelos, W. Joosen, C. Kruegel, F. Piessens, and G. Vigna, "Cookieless monster: Exploring the ecosystem of web-based device fingerprinting," in Proc. IEEE Symp. Security and Privacy (S&P), 2013, pp. 541–555.
- [6] P. Laperdrix, W. Rudametkin, and B. Baudry, "Beauty and the beast: Diverting modern web browsers to build unique browser fingerprints," in Proc. IEEE Symp. Security and Privacy (S&P), 2016, pp. 828–842.
- [7] T. Böttger, F. Cuadrado, G. Tyson, I. Castro, and S. Uhlig, "An empirical study of the cost of DNS-over-HTTPS," in Proc. ACM Internet Measurement Conf. (IMC), 2019, pp. 316–323.
- [8] M. A. Bashir and C. Wilson, "Diffusion of user tracking data in the online advertising ecosystem," Proc. Privacy Enhancing Technologies, vol. 2018, no. 4, pp. 85–103, 2018.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)