Cyber Guardian 360: A Web-Based Cybersecurity Toolkit for Real-Time Security Analysis

Mrs. R. Devika[1], S. Tirumalesh[2], Maturi Jaswanth Sai Madhu Mohan[3], V. Srinadh[4], K. Mohan[5]

[1]*Assistant Professor, Dept. of Cyber Security, Dhanalakshmi Srinivasan University, Tiruchirappalli, Tamil Nadu, India*

[2,3,4,5]*Dept. of Cyber Security, Dhanalakshmi Srinivasan University, Tiruchirappalli, Tamil Nadu, India*

***Abstract*—The rapid advancement of digital technologies has significantly increased the demand for accessible and practical cybersecurity tools. Traditional penetration testing environments such as Kali Linux require complex installation, high computational resources, and technical expertise, making them less suitable for beginners and educational use. This paper presents Cyber Guardian 360, a web-based cybersecurity toolkit designed to provide real-time security analysis through API-driven execution. The proposed system integrates multiple security functionalities such as network scanning, OSINT analysis, and reconnaissance directly within a browser environment. Unlike simulated platforms, Cyber Guardian 360 executes real tools through secure APIs, returning accurate, live outputs. The system is built on a React and Tailwind CSS frontend, supported by Supa base PostgreSQL for cloud-based data management. With over 21 integrated security tools, the platform addresses the limitations of existing tools by offering browser-based accessibility, multi-interface interaction, and real-time execution.**

***Index Terms*—cybersecurity, web-based toolkit, penetration testing, real-time analysis, OSINT, API-driven, network scanning, Nmap, Supa base, React**

I. INTRODUCTION

With the increasing reliance on digital systems, cybersecurity has become a fundamental aspect of modern computing environments. Organizations, educational institutions, and individuals require efficient tools to identify vulnerabilities, perform reconnaissance, and analyze potential threats. Traditional cybersecurity platforms such as Kali Linux and Parrot OS provide powerful tools but involve complex setup procedures, high system requirements, and steep learning curves. These limitations create barriers for students and beginners who aim to learn cybersecurity concepts practically. Additionally, most web-based cybersecurity tools available today either provide limited functionality or rely on simulated outputs, which do not accurately represent real-world security scenarios.

Cyber Guardian 360 addresses these challenges by providing a fully web-based platform capable of executing real security tools through API integration. The system targets students, security professionals, and bug bounty hunters who require practical, accurate, and efficient cybersecurity capabilities.

The platform features a multi-interface design that supports both beginners and advanced users. Tool Cards provide guided interaction for novice users, while a Terminal Emulator enables command-based execution for experienced professionals. A File Viewer module allows users to manage and review stored scan outputs efficiently.

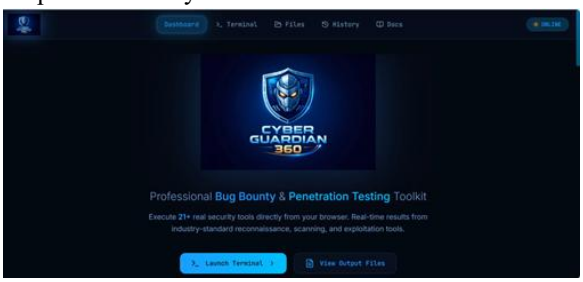

Figure 1: Cyber Guardian 360 – Home Dashboard

II. LITERATURE REVIEW

Several studies have explored different approaches to improving cybersecurity tool accessibility and effectiveness. Offensive Security (2019) introduced Kali Linux as a comprehensive penetration testing platform containing a wide range of tools. Although powerful, it requires installation on dedicated systems

or virtual machines, limiting accessibility for beginners [1].

Faletra et al. (2020) developed Parrot Security OS, which offers enhanced privacy features and lightweight performance. However, it still depends on systemlevel deployment and lacks web-based accessibility [2]. Pentest-Tools.com (2021) provides a browser-based vulnerability scanning platform. Despite this, its functionality is limited and many advanced features are restricted behind subscription models [3].

The OWASP Foundation (2023) developed the Web Security Testing Guide, establishing standards for web application security testing [4]. Supabase (2024) offers robust PostgreSQL cloud database services with RowLevel Security [5]. React (2024) continues to be a leading frontend framework enabling responsive, componentbased interfaces [6]. Recent IEEE research (2023) highlights the growing need for integrated, cloud-based cybersecurity platforms [7].

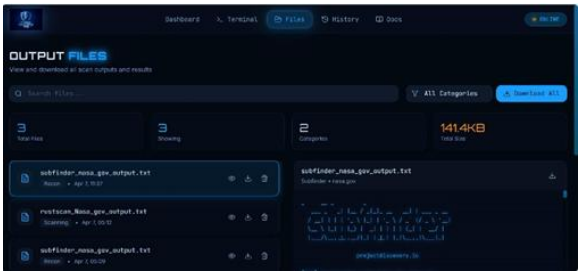

Fig. 3. Files Tab – Sub finder Output for nasa.gov (3082 Subdomains)

III. EXISTING SYSTEM

The existing landscape of cybersecurity tools presents several limitations that reduce their accessibility and effectiveness for beginners and educational users.

A. Installation-Based Tools

Traditional cybersecurity tools such as Kali Linux and Parrot OS require installation and configuration on local systems or virtual machines. This increases complexity and limits portability across different computing environments.

B. Limited Web-Based Platforms

Existing web-based platforms provide only basic scanning capabilities and often restrict advanced features through subscription models, limiting their educational and professional utility.

C. Simulated Output Systems

Many educational tools rely on simulated outputs rather than real execution, reducing effectiveness in practical learning scenarios and failing to prepare users for real-world security environments.

D. Hardware Dependency

Most tools require high-performance systems with significant RAM and storage, making them unsuitable for low-end devices commonly used in educational settings.

E. Lack of Integrated Workflow

Existing systems do not provide an integrated workflow that combines tool execution, result storage, and output management within a single platform.

IV. PROPOSED METHODOLOGY

The Cyber Guardian 360 system is designed on a modular, API-driven architecture that enables real-time execution of security tools directly within a browser environment.

A. Real-Time API-Based Execution

The proposed system integrates secure APIs to execute cybersecurity tools such as Nmap, Whois, Subfinder, Amass, Sublist3r, Waybackurls, GAU, and OSINT scanners. When a user initiates a scan, the request is processed through the API, and real-time results are returned directly to the browser interface.

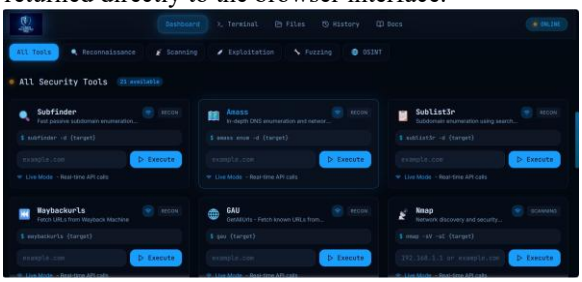

Figure 2: Security Tools Dashboard – 21 Integrated Tools

B. Multi-Interface Design

The system provides three execution modes: Tool Cards for guided interaction, a Terminal Emulator for command-based execution, and a File Viewer for managing stored outputs. This ensures usability for both beginners and advanced users.

C. Cloud-Based Data Storage

All execution results are stored in a secure cloud database using Supabase PostgreSQL. Row-Level Security ensures data privacy and controlled access for all users.

D. Modular Architecture

The system follows a three-tier architecture comprising the frontend interface, the API execution layer, and the cloud storage backend. This modular design allows independent scaling and maintenance of each component.

V. DEPLOYMENT

The Cyber Guardian 360 system is deployed as a cloud-based web application accessible through modern browsers. The frontend is built using React and Tailwind CSS, providing a responsive interface across devices. User requests are processed through an application layer that communicates with external APIs for tool execution. The results are stored in the Supa base database and displayed in real time.

The system supports scalable deployment using cloud infrastructure, enabling multiple users to perform simultaneous operations. Continuous updates and API integrations allow the platform to expand its capabilities over time.

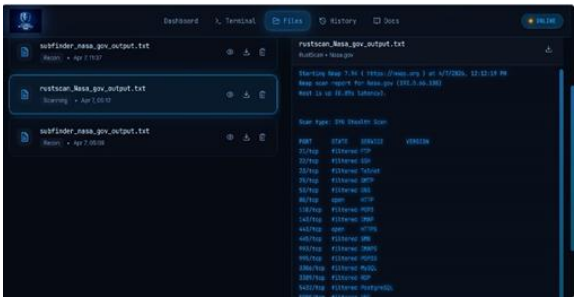

Figure 4: RustScan – Port Scan Results for Nasa.gov

VI. CHALLENGES

Despite its advantages, the system faces several challenges. API dependency is a critical factor, as the system relies on third-party services for real-time execution. Any downtime or limitation in API usage may affect overall platform performance.

Handling large volumes of requests may lead to latency issues, especially during peak usage. Additionally, realtime execution introduces security concerns that require strict validation and monitoring to prevent misuse. Ensuring accurate results across different tools and maintaining consistency in output formats are also ongoing challenges.

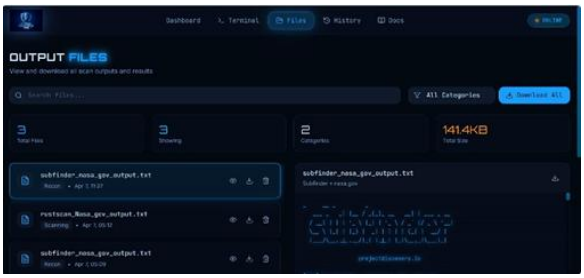

Figure 5: Output Files View – 3 Files, 2 Categories,1414KB

VII. CONCLUSION AND FUTURE WORK

Cyber Guardian 360 presents a modern approach to cybersecurity tool accessibility by integrating real-time API-based execution within a web platform. The system eliminates the need for installation and provides accurate, real-world outputs, enhancing practical learning and professional usability.

The platform successfully combines accessibility, performance, and security into a unified system suitable for students, educators, and professionals. Future work may include AI-based automated vulnerability detection, expansion of supported tools, development of a real-time monitoring dashboard, and collaborative workspace features for team-based penetration testing.

REFERENCES

[1] Kali Linux Documentation, Kali Linux, 2019.
[2] Parrot Security OS, Parrot Security OS, 2020.
[3] Pentest-Tools.com Online Vulnerability Scanner, Pentest-Tools.com, 2021.
[4] OWASP Foundation, “Web Security Testing Guide,” 2023. OWASP Web Security Testing Guide
[5] Supabase Documentation – PostgreSQL Security, Supabase, 2024.
[6] React Documentation, React, 2024.
[7] IEEE, “Research on Web-Based Cybersecurity Platforms,” 2023.