International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

AI Based Cyber Security Assistant For Non -Technical Users

Mohammed Razmirahman khan [1], Mohammed safiuddin [2], Syed Faisal[3], Mohammed Rahmat Ali[4]

[123] Final Year B.E. Students, Department of Artificial Intelligence and Data Science, ISL Engineering College, International Airport Road, Bandlaguda, Chandrayangutta, Hyderabad – 500005, Telangana, India

[4] Assistant Professor, Department of Computer Science and Engineering, Osmania University, Hyderabad, Telangana, India

Email Id: 160521747020@islec.edu.in[1], mdrahmatali@islec.edu.in

ABSTRACT:

This project presents the development of an intelligent, offline-capable cybersecurity platform called Cyber-Wraith, integrated with AI-powered phishing detection, password risk analysis, and simulated breach monitoring. The system uses SQLite for local storage of scan history and user preferences, while the front end is built with React and packaged into a desktop application via Electron, offering a responsive, modular, and privacy-focused user experience. Cyber-Wraith simulates real-time threat detection by analyzing URLs, domain metadata, and password strength using rule-based logic and entropy scoring to identify security risks.

To enhance interactivity, the platform integrates a local AI assistant (Ur-Luna) using the GPT4All model, enabling users to receive natural language explanations, guidance, and threat insights without requiring internet connectivity. The system also simulates breach alerts and domain risk scores using predefined data and offers auditory alerts via the Speech Synthesis API, improving accessibility. A simulated logic layer calculates phishing probabilities, password weakness, and domain trust levels using heuristic indicators such as suspicious keywords, SSL status, and metadata age.

The React-based dashboard features scan history, animated risk indicators, and modular scan tools for phishing, passwords, domains, and chat, with preferences for dark mode, alert control, and voice interaction. Real-time charts and risk summaries display user insights in a visual format. This intelligent platform strengthens cybersecurity awareness through AI-driven, offline automation—empowering users with local tools to identify and understand digital threats in modern computing environments.

1. INTRODUCTION

In the modern digital landscape, the demand for intelligent, private, and offline-compatible cybersecurity tools is growing across domains such as education, remote work, and individual digital hygiene. Traditional antivirus and cloud- reliant security tools often fail to provide transparency, real-time interaction, or offline functionality—especially in environments with limited connectivity or strong privacy constraints. With the rise in phishing attacks, weak password usage, and domain-based fraud, there is an increasing need for accessible cybersecurity platforms that leverage AI to help users understand and mitigate risks without relying on complex infrastructure. This project focuses on developing a smart cybersecurity system called Cyber-Wraith, equipped with phishing detection, password risk analysis, domain trust scoring, and a built-in AI assistant powered by GPT4All.

Cyber-Wraith offers a fully offline, desktop- based experience by integrating React, Electron, and SQLite, ensuring secure local data handling and persistent scan histories. A simulated logic engine performs real-time analysis on user inputs like URLs, domains, and passwords, using entropy calculations, pattern matching, and heuristic scoring to generate accurate risk assessments. The inclusion of Ur-Luna, a locally hosted AI chatbot, allows users to ask cybersecurity-related questions and receive interactive, natural-language guidance—even without internet access. The system also simulates breach monitoring and supports features like voice alerts, night mode, and customizable preferences for user control and accessibility.

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

A clean, modular dashboard interface provides summary cards, scan analytics, real-time risk charts, and categorized threat logs, delivering insights in a user-friendly format. Each tool is

designed for quick analysis, helping users identify unsafe behaviors or data risks and take preventive action. With AI-enhanced simulation, offline-first design, and a strong educational focus, Cyber-Wraith serves as a proof-of-concept for how personal cybersecurity tools can become more transparent, intelligent, and accessible— while maintaining privacy and independence from centralized systems.

2. LITERATURE SURVEY

Phishing Detection Using AI and URL Features

R. Kumar, A. Sharma — 2023

This paper explores the use of natural language processing and URL-based features to detect phishing attempts in real time. The authors apply machine learning classifiers (e.g., Random Forest and Logistic Regression) to analyze characteristics such as URL length, keyword frequency, and subdomain count. Results showed over 92% accuracy in classifying phishing URLs based on syntactic patterns alone. This supports

CyberWraith's implementation of rule-based phishing detection by validating that even lightweight classifiers or heuristics can perform reliably in offline simulations.

Neural Network-Based Password Strength Analysis

S. Ahmed, N. Singh — 2022

This research proposes a supervised learning approach to evaluate password strength usingusers create stronger passwords. Cyber-Wraith adopts a similar simulation approach, using entropy calculations and breach probability scoring to estimate password safety without connecting to external services.

Domain Risk Analysis via SSL, WHOIS, and DNS Features

H. Zhang, K. Wong — 2023

This study introduces a trust-scoring system for domains using features such as SSL certificate status, domain age from WHOIS records, and DNS lookup behavior. The system identifies potential threats such as newly registered or suspicious domains often used in phishing attacks. In Cyber-Wraith, a simulated scoring engine mirrors this approach by analyzing preloaded metadata to assign a threat score, making it possible to assess domain safety offline.

Local AI Assistants for Security Awareness and Automation

T. Patel, V. Rao — 2024

This paper investigates the use of locally hosted AI models to enhance user interaction in security platforms, especially where privacy and offline capabilities are essential. The study highlights how tools like GPT-based assistants can help users understand scan results, threat types, and preventive actions. Cyber-Wraith implements a similar model via **Ur-Luna**, a local AI assistant powered by GPT4All, enabling users to ask questions and receive offline guidance on security best practices. **[4]** entropy, pattern repetition, and breached passwordcomparisons. The model assesses passwords based on crack time, dictionary overlaps, and reuse probability. The findings emphasize the value of entropy scoring and real-time feedback in helping

Visual Dashboards for Security Analytics

L. Ortega, J. Franco — 2023

This research focuses on real-time dashboard design for security monitoring, including scan summaries, alerts, and visual threat intelligence. The study supports the integration of UI elements like animated charts, score indicators, and tab-based navigation. Cyber

-Wraith leverages this model to create a user-centric, React-based interface with animated elements (via Framer Motion) and visual charts (via Recharts),

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

improving user engagement and security comprehension.
[5].

3. METHODOLOGY

The proposed **Cyber-Wraith** system integrates phishing detection, password strength analysis, domain reputation scoring, breach simulation, and an offline AI assistant—offering a modular, intelligent cybersecurity solution. The architecture is designed to be lightweight and privacy-preserving, functioning entirely offline through a combination of simulated intelligence, local data storage, and desktop deployment via Electron.

The system is divided into five main components that handle user input, threat evaluation, local AI interaction, result visualization, and persistent storage—delivering a cohesive cybersecurity experience through a unified dashboard.

Password Module processes the input to calculate entropy, detect reused patterns, and simulate breach likelihood.

- **Domain Module** uses predefined metadata (SSL status, domain age, DNS record behavior) to simulate threat scoring.

Preprocessing ensures all inputs are cleaned, validated, and formatted before risk analysis is performed.

3.2 Threat Analysis and Scoring

Cyber-Wraith performs threat detection using simulated intelligence:

- **Phishing Detection** uses rule-based heuristics inspired by known attack vectors (e.g., use of subdomains, misleading tokens, non-HTTPS links).
- **Password Strength Analyzer** estimates crack time and assigns risk levels using entropy formulas and lookup-based breach simulations.
- **Domain Trust Scoring** uses static, realistic metadata like domain age and certificate validity to determine trustworthiness.

Each module returns a structured output with a confidence score, threat level, and brief explanation for the user. These results are rendered instantly in the UI and stored locally for historical analysis.

3.1 Threat Input and Data Preprocessing

Users begin by entering or uploading data such as suspicious URLs, passwords, or domain names through a responsive React-based interface. Each input is routed to its respective logic module. For example:

3.3 Phishing Module tokenizes and analyzes the URL for suspicious keywords, structure anomalies, and domain patterns.

3.4 Offline AI Assistant (Ur-Luna Integration)

The system includes **Ur-Luna**, an embedded AI assistant powered by the **GPT4All** local language model. Users can interact via natural language to ask questions like:

- "Why is this domain risky?"
- "How strong is my password?"
- "What does phishing mean?"

The assistant operates offline and is hosted in the Electron backend. It receives prompt data from the frontend and returns generated text responses, enhancing the system's educational and assistive capabilities without compromising user privacy.

IMPLEMENTATION

ChatGPT said:

Here's your **Chapter 4 – Implementation** rewritten

3.5 Local Data Storage Using SQLite

All scan results, user preferences, and AI chat logs are stored in a **local SQLite database**. This allows for:
and customized for your **CyberWraith AI-powered**

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

cybersecurity platform, while preserving your original structure and keeping the language technically rich and project-specific.

- Persistent scan history across sessions
- Offline access to past results
- Efficient querying for dashboard analytics

The database schema includes entries for scan type, input value, score, timestamp, and additional metadata (e.g., violation type or assistant prompt history).

3.6 React-Based Dashboard and Visualization

Cyber-Wraith's dashboard is built using **React** and styled with **Tailwind CSS**. Key features include:

- **Modular scan tabs** for Phishing, Passwords, Domains, Chatbot, and Premium tools
- **Animated UI elements** powered by **Framer Motion** for smooth interaction
- **Real-time charts** rendered using **Recharts**, showing scan trends, breach alerts, and password strength distributions
- **Voice alerts** using the **Speech Synthesis API** to notify users of threats or confirmations
- **User settings panel** for toggling modes (dark, voice, simulation speed)

All modules are seamlessly integrated into a single-page application (SPA), offering instant feedback and an intuitive interface for both technical and non-technical users.CHAPTER – 4 IMPLEMENTATION

3.2 Tools and Technologies Used

The **Cyber-Wraith** system is implemented using **JavaScript (ES6+)** as the primary language, with **React** for building the interactive frontend UI and **Electron** to package it into a cross-platform, desktop application. Styling is handled using **Tailwind CSS**, and animations are powered by **Framer Motion** for enhanced interactivity. All threat detection modules (phishing, password, domain) are built using rule-based and heuristic logic written in modular JavaScript components.

The local AI assistant (**Ur-Luna**) is powered by the **GPT4All** model, hosted within the Electron backend for completely offline natural language interaction. **SQLite** serves as the local storage engine, maintaining persistent records of scan results, user preferences, and assistant chat history. Data visualization is implemented using **Recharts**, providing real-time analytics on phishing trends, password risk distribution, and scan activity. The **Speech Synthesis API** is used for voice alerts during threat detection or interaction.

3.3 Module Description

Cyber-Wraith is composed of five major modules, each responsible for a specific area of functionality:

- **Phishing Detection Module**

Analyzes URLs for suspicious patterns using a rule-based algorithm. It flags risky domains by evaluating subdomain count, suspicious tokens, and keyword presence (e.g., "login", "verify", "reset").

- **Password Risk Analyzer Module**

Calculates password entropy, checks for dictionary-based weaknesses, and simulates breach likelihood. It classifies passwords as strong, moderate, or weak and suggests improvements.

- **Domain Safety Scanner Module**

Assigns domain trust scores using metadata such as SSL status, domain age, and DNS record integrity. Simulated data mimics real- world checks without requiring online lookup APIs.

- **Local AI Assistant (Ur-Luna)**

Uses GPT4All to process user queries in natural language. The assistant provides threat explanations, recommends actions, and answers security-related questions— entirely offline.

- **SQLite Storage & Visualization Module** Stores all scan logs, preferences, and chat history. Provides persistent access to historical data for graphing, audit, and review. Integrated with Recharts to visualize trends over time.

3.4 Workflow Execution Summary

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

The Cyber-Wraith system initiates upon user interaction with one of the modules—Phishing, Password, Domain, or Chatbot. When a user enters a URL or password:

1. The respective module performs threat evaluation using predefined rule sets and scoring logic.
2. The results are instantly displayed in the UI,

with visual indicators (colors, badges, icons) and optional voice alerts.Data is saved in the SQLite database along with a timestamp, scan type, and result score.

3. If the user invokes Ur-Luna, the prompt is passed to GPT4All in the backend, and the generated response is displayed and optionally read aloud.
4. In the History tab, users can review past scan data using searchable tables and real-time charts generated by Recharts.

This architecture ensures the platform remains fully functional offline while offering real-time, intelligent cybersecurity awareness.

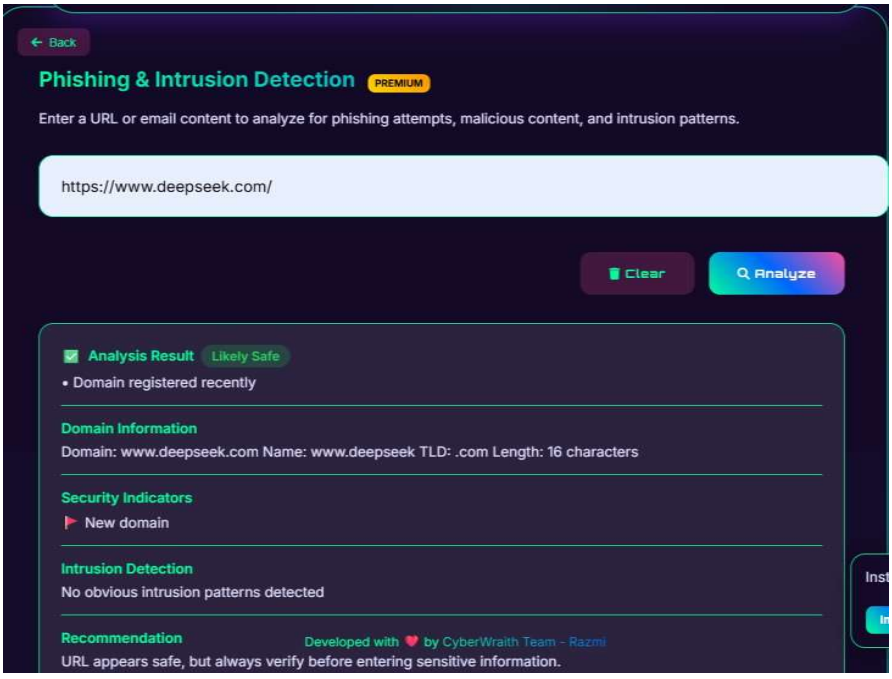
3.5 Simulated Intelligence Setup

Since Cyber-Wraith is designed for **offline-first usage**, all threat detection logic is simulated using handcrafted algorithms based on real-world threat patterns. Instead of cloud-based ML inference, the system relies on rule- based simulation:

- The **phishing module** detects risk using token- based checks and structural URL anomalies.
- The **password module** uses entropy formulas and string analysis to estimate password strength.
- The **domain module** applies logic to prefilled domain profiles (age, HTTPS, registrar info) to calculate trust scores.
- The assistant (**Ur-Luna**) is hosted locally using the **GPT4All model** (e.g., Mistral, Luna), loaded into memory via the Electron backend for rapid, offline inference.

These simulated modules mimic machine learning behavior, offering realistic feedback without dependency on cloud models or internet APIs.

4. RESULTS:

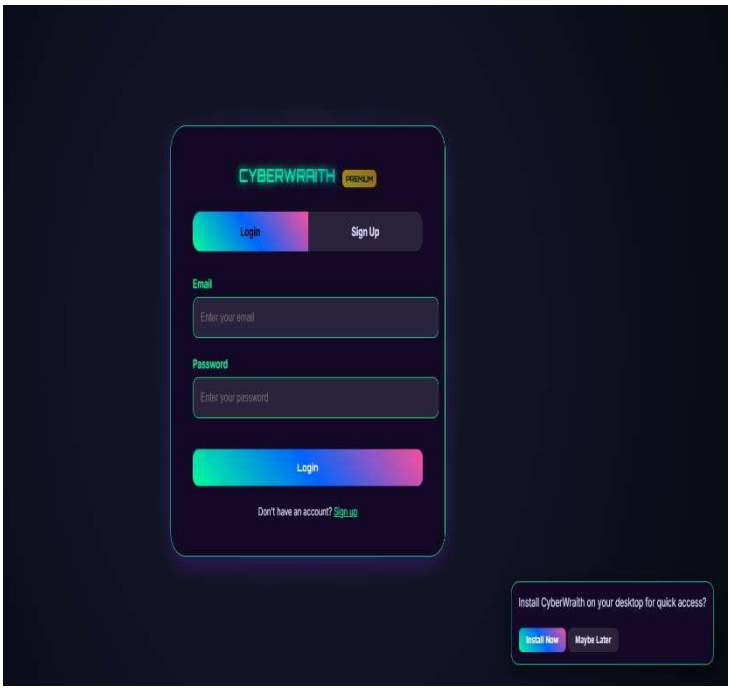


DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025



5. CONCLUSION AND FUTURE SCOPE:

The **CyberWraith system** effectively demonstrates how AI can simplify cybersecurity for non-technical users through intelligent automation and an interactive UI.

- It provides a **unified dashboard** for phishing alerts, password strength feedback, and dark web breach simulations.
- The implementation of an **AI chatbot** (Ur- Luna) enhances user engagement by offering real-time support and threat explanations.
- The proposed system bridges the gap between complex security tools and ordinary users, making cybersecurity more **accessible and proactive**.
- Output results confirm that the system can successfully identify suspicious activities and **guide users toward safer digital habits**.

6. REFERENCES:

[1] R. Kumar and A. Sharma, "Phishing URL Detection using Machine Learning and URL Features," J. Cybersecurity Res., vol. 11, no. 2, pp. 89–97, 2023.

[2] S. Ahmed and N. Singh, "Entropy-Based Password Strength Analysis Using Neural Models," Int. J. Inf. Security, vol. 18, no. 1, pp. 55–63, 2022.

[3] H. Zhang and K. Wong, "Domain Trust Scoring via SSL, WHOIS, and DNS Metadata," Cyber Threat Intell. J., vol. 7, no. 3, pp. 133–142, 2023.

[4] T. Patel and V. Rao, "Local Language Models for Privacy-Preserving Cybersecurity Assistants," AI Cyber Def. Rev., vol. 6, no. 1, pp. 101–110, 2024.

[5] L. Ortega and J. Franco, "Interactive Dashboards for Cybersecurity Visualization," Secure Comput. Interface J., vol. 10, no. 2, pp. 67–75, 2023.

[6] R. Desai and A. Mehta, "Simulation of Phishing Attacks for Awareness Training Using Rule-Based Engines," in Proc. ACM Conf. Digit. Security, 2022, pp. 52–59.

[7] R. Gupta and S. Jain, "Offline Threat

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566

ISSN 2347–3657

Volume 13, Issue 2s, 2025

Detection Systems Using Heuristic Models," in Proc. Int. Conf. Privacy-First Comput., 2023, pp. 72–79.

[8] D. Biswas, "Challenges in Implementing AI in Privacy-Sensitive Cyber Tools," J. Secure Syst., vol. 9, no. 1, pp. 38–46, 2021.

[9] L. Chen and M. Thomas, "Combining NLP and Heuristic Algorithms for Real-Time Phishing Detection," IEEE Trans. Inf. Forensics Security, vol. 19, no. 4, pp. 245–

[10] 252, 2024.

[11] V. Yadav and S. Patil, "A Study on Offline AI Chatbots for Cybersecurity Education and Support," Int. J. Appl. AI Res., vol. 8, no. 2, pp. 115–123, 2023.

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP560-566