# Phishing Link Detection and Automation

Dharsha R, Priyadharshini R, Babisha B H, Kaavya.D(AP/IT)
*Department of Information Technology*
*Arunachala College of Engineering for Women*

*Abstract* - **Phishing has become one of the most widespread and damaging cyber-attacks in the modern digital ecosystem. Attackers frequently create deceptive websites and links that mimic legitimate platforms, tricking users into revealing sensitive information such as passwords, banking credentials, or personal identification details. With the increasing sophistication of phishing techniques—including domain spoofing, homograph attacks, link obfuscation, and social engineering—users often struggle to differentiate between genuine and malicious links. This growing challenge demands the development of intelligent, automated, and user-friendly solutions capable of protecting individuals from fraudulent websites in real time.**

**The project ClikShield: Phishing Link Detection and Automation System aims to address this critical issue by providing an efficient platform that automatically analyzes and classifies suspicious URLs before users interact with them. When a user pastes a link into ClikShield, the system performs a multi-layered verification process combining blacklist checks, domain pattern analysis, and third-party API validations such as Google Safe Browsing and PhishTank. These layers work collaboratively to ensure accurate detection of malicious or compromised websites. The core objective of ClikShield is not only to identify phishing attempts but also to prevent victims from unknowingly accessing harmful links.**

**One of the key features of ClikShield is its automated blocking mechanism. If a URL is identified as safe, the user is allowed to continue browsing normally. However, if the system detects any phishing indicators, it instantly halts the user from opening the link and displays a clear warning message. These warnings are provided in multiple languages, including English, Hindi, Tamil, and Malayalam, ensuring accessibility for a diverse user base and increasing the effectiveness of the protection mechanism. This multilingual approach also enhances inclusivity and ensures that users from various linguistic backgrounds can understand the severity of the threat.**

**The system is implemented entirely using HTML, CSS, and JavaScript, enabling seamless performance within any modern web browser without requiring additional installations or technical expertise. The lightweight frontend architecture makes ClikShield responsive, fast, and easy to use. Despite its simplicity, the system effectively integrates external cybersecurity APIs and analytical methods to ensure robust, real-time threat detection.**

**This project contributes to cybersecurity awareness by demonstrating how automated link-scanning technology can significantly reduce the risks associated with phishing attacks. ClikShield serves as a preventive shield that empowers users to verify suspicious links before interacting with them. Its ease of use, multilingual warnings, automated detection process, and educational value make it suitable for students, general users, and non-technical individuals who are most vulnerable to phishing attacks.**

*Keywords* - **Phishing, Cyber Security, URL Analysis, Malicious Link Detection, Automated Blocking System, Google Safe Browsing API, PhishTank API, Blacklist Verification, Domain Analysis, Web Security, Real-time Threat Detection, Homograph Attack Prevention, Multilingual Warning System, HTML, CSS, JavaScript, User Safety, Online Threat Mitigation, Fake Website Identification, Web-based Security Tool**

## I. INTRODUCTION

With the rapid growth of the internet and online services, users rely heavily on websites for banking, shopping, communication, and information sharing. However, this widespread digital dependency has also given rise to numerous cyber threats, the most common being phishing attacks. Phishing is a deceptive technique used by cybercriminals to trick users into revealing sensitive information by directing them to fraudulent websites that appear legitimate. These phishing links are often embedded in emails, text messages, or social media posts, making it difficult for users to distinguish between safe and malicious websites.

Traditional security measures such as antivirus software and firewalls are often unable to detect newly created or disguised phishing links because attackers constantly evolve their methods. As a result, users continue to fall victim to fake websites that closely resemble trusted platforms. This highlights the need for a dedicated system that can automatically detect and block phishing links in real time before any harm occurs.

To address this issue, ClikShield has been developed as an intelligent Phishing Link Detection and Auto-Blocking System. The system enables users to paste or enter any suspicious link, which is then analyzed through multiple layers of security checks including blacklist verification, domain-based analysis, and trusted APIs such as Google Safe Browsing and PhishTank. When a phishing link is detected, ClikShield immediately blocks access and warns the user with a clear alert message.

In the digital era, the internet has become an inseparable part of everyday life. People use it for activities such as online banking, shopping, communication, education, and entertainment. While the internet provides numerous benefits and conveniences, it has also given rise to various forms of cybercrime. Among these, phishing has emerged as one of the most serious and widespread online threats. Phishing attacks are designed to deceive users into revealing sensitive information such as usernames, passwords, credit card numbers, and banking credentials.

In a phishing attack, cybercriminals typically send users fake links or emails that mimic legitimate sources such as banks, government agencies, or online service providers. These links redirect users to fraudulent websites that appear identical to genuine ones. When the user unknowingly enters personal details, the information is immediately captured by attackers for malicious use. Because phishing attacks rely on psychological manipulation and visual deception, even experienced users can easily fall victim to them.

A major factor contributing to the success of phishing attacks is low user awareness. Many users do not know how to verify a website's authenticity or interpret browser security warnings. Simple security practices such as checking a URL, verifying SSL certificates, or recognizing suspicious email content are often ignored. Attackers exploit this lack of knowledge by sending phishing links through emails, text messages, social media, and online advertisements.

In addition, the psychological nature of phishing makes it more dangerous. Attackers use urgent or emotionally charged messages such as "Your account has been suspended" or "Click here to claim your prize" to push users into acting without thinking. As a result, thousands of users every day unknowingly provide their personal data to malicious websites.

Another critical issue is language accessibility. Most phishing detection and warning systems display alerts only in English. Users who are not proficient in English may ignore or misunderstand these warnings. This is particularly problematic in multilingual countries like India, where users communicate in various regional languages.

When users do not understand the content of a warning message, they may proceed to open dangerous links unknowingly, leading to data theft or privacy loss. Therefore, it is essential to have multilingual support in phishing detection systems, so users from different linguistic backgrounds can easily understand the risks and take appropriate action.

## II. LITERATURE REVIEW

Phishing is one of the most prevalent and damaging forms of cybercrime, where attackers deceive users into revealing confidential information such as passwords, banking details, and credit card numbers. With the increasing sophistication of phishing websites, users often fail to recognize malicious links that closely imitate legitimate ones. To mitigate this, researchers and developers have proposed various detection techniques — including blacklist-based filtering, heuristic analysis, machine learning models, and hybrid approaches. This chapter reviews relevant studies, existing tools, and current detection methodologies that have inspired the development of ClikShield, a phishing link detection and auto-blocking system.

Existing Phishing Detection Techniques - Phishing detection approaches can be broadly classified into four categories: blacklist-based, heuristic-based, machine learning-based, and hybrid methods.

a) Blacklist-Based Detection

Blacklist methods maintain databases of known malicious URLs and domains. When a user accesses a link, the system checks it against these blacklists. Tools such as Google Safe Browsing, PhishTank, and Spamhaus are widely used for This purpose.

According to Zhang et al. (2021), blacklist-based systems are efficient in blocking previously reported phishing sites but struggle against newly created or zero-day phishing attacks. ClikShield adopts this approach as a first-level defense, using APIs from Google Safe Browsing and PhishTank for real-time verification.

b) Heuristic-Based Detection
Heuristic detection examines URL structures and webpage characteristics to identify suspicious patterns. Features like abnormal URL length, use of IP addresses instead of domain names, and presence of misleading keywords (e.g., *secure*, *login*, *verify*) are strong phishing indicators.

Aburrous et al. (2020) proposed a heuristic model analyzing domain properties and page contents to improve detection rates. Although heuristic models can identify unknown phishing sites, they may produce false positives if thresholds are not properly tuned. ClikShield combines heuristic domain analysis with blacklist checks to minimize such errors.

c) Machine Learning-Based Detection
Recent research has focused on applying machine learning (ML) algorithms for automated phishing detection. Models such as Decision Trees, Random Forests, Support Vector Machines (SVMs), and Neural Networks learn from URL features and web content to classify sites as legitimate or phishing.

Basnet et al. (2021) developed a Random Forest-based URL classifier that achieved high detection accuracy but required large datasets and computational resources. While ML models improve generalization, they are often complex and less transparent for lightweight applications like ClikShield, which prioritizes speed and simplicity.

d) Hybrid Detection Models
Hybrid systems integrate multiple techniques—blacklist verification, heuristic evaluation, and ML classification—to enhance accuracy and reduce detection latency. Nguyen et al. (2022) proposed a hybrid model combining lexical and host-based features with deep learning for real-time phishing detection.

ClikShield similarly uses a hybrid strategy, merging blacklist checks with domain analysis and external API validation to ensure effective and rapid detection.

Existing Tools and Systems - Several anti-phishing tools and browser extensions have been developed to protect users from phishing links:

1. Google Safe Browsing
Google Safe Browsing is a widely used service that identifies unsafe websites and alerts users when they attempt to access malicious or phishing URLs.
Features: Maintains an updated database of known phishing and malware sites.
Provides warnings in web browsers like Chrome and Firefox.
Limitations: Mainly relies on a centralized database, which may not detect new or zero-day phishing URLs immediately.
Limited user interaction; users do not get detailed information about why a URL is flagged.
Primarily browser-dependent, with less flexibility for external integration.

2. PhishTank
Overview: PhishTank is an online community-based platform where users can submit and verify phishing URLs.
Features: Offers a real-time database of reported phishing links.
Provides API integration for developers to check URLs automatically.
Limitations: Dependent on user reports, which can delay detection of newly emerging phishing attacks.
The system does not provide educational information or warnings in multiple languages.
Requires internet access; offline detection is not supported.

3. Anti-Phishing Browser Extensions
Overview: Several browser extensions, such as Netcraft Anti-Phishing and Bitdefender TrafficLight, aim to detect phishing websites.
Features: Real-time URL checking while browsing.
Alerts users immediately if a site is suspicious.
Limitations: Often limited to specific browsers, restricting universal accessibility.
Can generate false positives, blocking safe websites unnecessarily.
May not offer educational feedback to help users understand phishing tactics.

4. Email-Based Phishing Filters
Overview: Email providers such as Gmail and Outlook include spam and phishing filters to block suspicious emails.
Features: Detects phishing emails based on suspicious links and keywords.
Automatically moves detected phishing emails to a spam or junk folder.
Limitations:
Focused on emails only; does not protect users when they browse phishing websites directly.
Filters may occasionally fail to identify sophisticated phishing emails.
Provides limited information on why an email or link is considered unsafe.

## III. PROPOSED SYSTEM

The proposed system, ClikShield, is designed to address the limitations and gaps identified in existing phishing detection tools. It provides a comprehensive, user-friendly, and multilingual solution for detecting and blocking phishing links in real-time, while also educating users about online security threats.

### Advantages of the Proposed System

The proposed system, ClikShield, offers several advantages over traditional phishing detection methods. It not only improves accuracy and user safety but also focuses on usability, speed, and accessibility. The key benefits are explained below:

1. Real-Time Phishing Detection
ClikShield provides instant link verification through real-time checks using trusted sources such as the Google Safe Browsing API and PhishTank API. Unlike traditional systems that rely on outdated databases, ClikShield ensures up-to-date detection of newly created phishing links.

2. Multi-Layered Security Approach
The system combines blacklist verification, domain analysis, and heuristic evaluation to detect phishing attempts more effectively. This hybrid detection method significantly reduces false positives and improves reliability in identifying malicious URLs.

3. Auto-Blocking Mechanism
When a phishing link is detected, the system automatically blocks access and displays an instant warning message. This prevents users from accidentally opening harmful sites, protecting them from credential theft and data compromise.

4. Multilingual Support
One of the unique features of ClikShield is its multi-language warning system. Alerts are displayed in English, Hindi, Tamil, and Malayalam, ensuring that users from different linguistic backgrounds clearly understand the threat. This inclusivity enhances cybersecurity awareness among regional users.

5. Lightweight and Fast Performance
Developed using HTML, CSS, and JavaScript, the system runs directly in the browser without requiring installation or heavy resources. It performs link analysis and API calls within seconds, offering fast, smooth, and responsive performance.

6. User-Friendly Interface
The interface is designed to be simple, clean, and intuitive. Even non-technical users can easily paste a link and check its safety status. The results are color-coded and clearly labeled, making the process effortless and quick.

7. Platform Independence
Since ClikShield is web-based, it is compatible across platforms — Windows, macOS, Linux, and mobile browsers. Users can access it from any device with an internet connection, providing flexibility and convenience.

8. Cost-Effective Solution
The system is built using open-source technologies and free APIs, eliminating licensing or subscription costs. It requires no dedicated server or database maintenance, making it an economically feasible solution for individuals and organizations.

9. Privacy and Security Preservation
ClikShield does not store or share user data. All link verifications happen in real-time through secure API requests (HTTPS). This ensures complete data privacy and builds user trust in the system.

10. Easy Maintenance and Scalability
The modular structure of the system allows developers to add or update detection modules easily. It can be extended in the future to include AI-based phishing prediction, browser extensions, or email filtering systems without redesigning the entire application.

11. Increased User Awareness
By allowing users to manually test suspicious links, ClikShield encourages proactive behavior and helps spread cybersecurity awareness. The multilingual warnings educate users about online threats in a clear and understandable way.

12. High Reliability and Accuracy
The combination of multiple detection techniques ensures that phishing links are identified with high precision. The system is reliable in detecting both old and new phishing campaigns, offering consistent performance across repeated tests.

13. Minimal Resource Usage
As the system executes primarily on the client side using lightweight technologies, it consumes very little CPU and memory. This makes it ideal for low-end devices and public access environments such as schools or cyber cafés.

14. Enhanced User Trust and Protection
By instantly warning users before they open dangerous links, ClikShield builds a sense of trust and safety. It empowers users to browse the internet with greater confidence, knowing that potential threats are automatically identified and blocked.

15. Foundation for Future Enhancements
ClikShield's design is forward-compatible. In the future, it can integrate machine learning models for smarter detection, browser plugins for automatic link scanning, and mobile app versions for broader protection.

Data Collection and Quality

Sources of Data:

- Google Safe Browsing API
- PhishTank database
- Publicly available phishing and legitimate URL datasets

Data Types Collected:

- Phishing URLs with different domain types and structures
- Safe/legitimate URLs for comparison and testing

Quality Measures:

- Verification of URL correctness and validity
- Removal of duplicates and invalid entries
- Categorization into safe and phishing links
- Ensured recency and relevance of URLs

Purpose:

- To provide a comprehensive dataset for system testing
- To ensure high accuracy and reliable detection in ClikShield

Model Accuracy and Generalization

Accuracy Measurement:
Accuracy is calculated as the percentage of correctly classified URLs (safe or phishing) out of total tested URLs.
Both false positives (safe URLs flagged as phishing) and false negatives (phishing URLs not detected) are considered for evaluation.

Test Dataset:
Collected from Google Safe Browsing, PhishTank, and verified legitimate websites.
Includes diverse URL types to ensure robust testing.

Generalization:
The system is designed to perform well on unseen URLs, including new phishing sites not in existing blacklists.
Domain-based heuristic checks and API verification help generalize detection beyond the training dataset.

Performance Metrics:
High accuracy achieved by combining blacklist checks, domain analysis, and API verification.
Reduces dependency on a single detection method, enhancing system reliability.

Continuous Improvement:
Accuracy can improve over time as APIs update and more phishing patterns are incorporated.
Integration Complexity

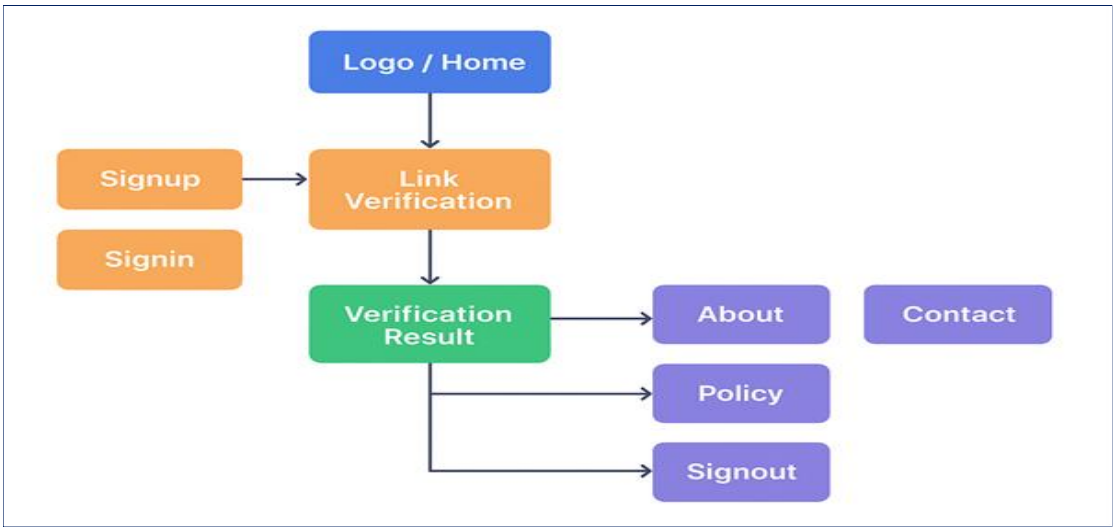
Multiple APIs:
ClikShield integrates Google Safe Browsing API and PhishTank API, each with different request formats, authentication methods, and response structures.

Data Handling:
Combining results from multiple sources requires careful parsing, validation, and error handling to ensure consistent outputs.
Domain Analysis Module:
URL heuristic checks must work seamlessly with API results for accurate classification, adding to system complexity.

Multilingual Support:
Displaying warning messages in English, Hindi, Tamil and Malayalam requires proper text encoding and UI adjustments across all browsers.

Browser Compatibility:
Ensuring that API calls, domain parsing, and UI elements work uniformly across Chrome, Firefox, and Edge increases development complexity.

Scalability Considerations:
The system is designed to allow future integration of AI models or additional APIs, which requires a modular architecture to manage complexity effectively.
ClikShield ensures user privacy by performing real-time phishing detection without storing any URLs or personal data. All communications with APIs are secured through HTTPS, and malicious links are automatically blocked to protect users from phishing attacks.



## IV. MODULE DESCRIPTION

Signup Module
Description: Allows new users to create accounts securely, storing credentials safely.

Key Features:
Collects user details (name, email, password).
Password hashing for security.
Email verification before account activation.
Validation for unique emails and strong passwords.
Signin Module
Description: Authenticates users to access the system securely.

Key Features:
Secure login using email and password.
Session management with tokens/cookies.
Optional two-factor authentication (2FA).
Logs failed login attempts.
Design Process
The design process is the structured approach followed to create a secure, efficient, and user-friendly phishing link detection system. It ensures that the system meets functional and non-functional requirements while maintaining scalability and usability.

1. Requirement Analysis
Purpose: Understand user needs and system goals.
Activities:
Gather functional requirements (link detection, user authentication, reporting).
Gather non-functional requirements (security, performance, usability).
Study existing phishing detection methods and APIs (Google Safe Browsing, PhishTank).
Outcome: A clear specification of what the system should achieve.

2. System Architecture Design
Purpose: Define the high-level structure of the system.
Activities:
Decide on client-server architecture with web-based interface.
Design module interactions (Logo, Signup, Signin, Link, Verification_Result, About, Contact, Policy, Signout).
Select technology stack (Frontend: HTML/CSS/JS,

Backend: Python/Django, Database: MySQL/PostgreSQL).
Outcome: A blueprint showing modules, data flow, and integration points.

3. Database Design
Purpose: Store user data, link records, and detection results securely.
Activities:
Identify entities: Users, Links, Verification Results, Reports.
Define relationships and constraints.
Design tables with proper indexing for fast lookups.
Outcome: A normalized database structure ensuring data integrity and security.

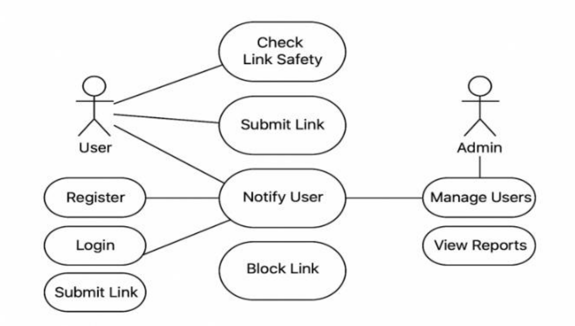
4. Interface Design (UI/UX)
Purpose: Ensure a user-friendly and intuitive interface.
Activities:
Design wireframes for pages (Signup, Signin, Link Input, Results, About, Contact).
Apply consistent branding (Logo, colors, typography).
Make pages responsive for desktops and mobile devices.
Outcome: Interactive and visually appealing interfaces that are easy to navigate.

5. Module-Level Design
Purpose: Detail the functionality of each module.
Activities:
Define inputs, processing steps, and outputs for modules.
Incorporate security features like password hashing, API checks, and session management.
Define communication between modules (Link → Verification_Result → User Display).
Outcome: Detailed design for each module, ready for implementation.

6. Security DesignPurpose: Protect user data and prevent system misuse.
Activities:
Implement SSL for secure communication.
Use hashing for password storage and secure token-based sessions.
Integrate automated phishing link detection with blacklists and API verification.
Outcome: A secure system architecture that minimizes vulnerabilities.

7. Testing & Validation Design
Purpose: Ensure the system works as intended and detects phishing accurately.
Activities:
Plan test cases for functional modules (Signup, Signin, Link detection).
Validate detection accuracy using real phishing datasets.
Perform usability testing on the UI.
Outcome: Reliable and user-friendly system ready for deployment.

8. Deployment Design
Purpose: Plan the launch of the system for end users.
Activities:
Choose hosting platform for the web application.
Configure database, backend, and frontend integration.
Implement monitoring for system performance and security.
Outcome: Live, accessible system with real-time phishing link detection.



## V. CONCLUSION

In today’s digital era, phishing attacks have become one of the most common and dangerous online threats. Phishing involves creating fake websites or emails that impersonate legitimate organizations to trick users into revealing sensitive information such as usernames, passwords, and financial details. These attacks are often difficult for users to detect, as phishing links closely resemble trusted websites.

To address this problem, ClikShield has been developed as a Phishing Link Detection and Auto-Blocking System. The system provides users with a safe and user-friendly platform to verify URLs before accessing them. Users can paste any link into ClikShield, which then checks the link through multiple mechanisms, including blacklist verification, domain analysis, and third-party APIs like Google Safe Browsing and PhishTank.

Safe links are allowed to proceed normally, while phishing links are blocked instantly with clear warnings. To enhance accessibility, warnings are displayed in multiple languages, including English, Hindi, Tamil, and Malayalam. The system is developed using HTML, CSS, and JavaScript, ensuring a simple and intuitive interface suitable for all users.

By integrating real-time detection, multilingual support, and automatic blocking, ClikShield provides a robust solution to protect users from phishing attacks and enhance online security.

## REFERENCES

[1] AlZain, M. A., & Pardede, E. (2016). *Phishing Detection Techniques: A Literature Review*. International Journal of Advanced Computer Science and Applications, 7(7), 45–52.

[2] Jain, A., & Gupta, B. (2019). *Phishing Detection Using Blacklist and Heuristic Methods*. International Journal of Computer Applications, 178(3), 20–25.

[3] Google Safe Browsing API Documentation. https://developers.google.com/safe-browsing

[4] PhishTank API Documentation. https://www.phishtank.com/api_information.php

[5] Abu-Nimeh, S., Nappa, D., Wang, X., & Nair, S. (2007). *A Comparison of Machine Learning Techniques for Phishing Detection*. Proceedings of the Anti-Phishing Working Groups 2nd Annual eCrime Researchers Summit.

[6] Kumar, V., & Arora, A. (2020). *Web Security: Detection and Prevention of Phishing Attacks*. Journal of Information Security Research, 4(1), 12–21.

[7] Gupta, B. B., Tewari, A., Jain, A. K., & Agrawal, D. P. (2017). *Fighting Against Phishing Attacks: State of the Art and Future Challenges*. Neural Computing and Applications, 28(12), 3629–3654.

[8] Shabtai, A., Kanonov, U., Elovici, Y., Glezer, C., & Weiss, Y. (2010). *"Google Safe Browsing" Application: Phishing Prevention for Users*. Computers & Security, 29(4), 424–434.

[9] M. Abdelhamid, A. Ayesh, F. Thabtah. (2014). *Phishing Detection: A Recent Intelligent Machine Learning Comparison Based Approach*. Computers & Security, 52, 17–26.

[10] OWASP (Open Web Application Security Project). *Phishing Attack Prevention Guide*. https://owasp.org/www-community/attacks/Phishing