IBM Innovation Center for Education

PHEMESOFT Software That Matters
TRUSTED DELIVERY PARTNER for IBM ICE

YENEPOYA
(DEEMED TO BE UNIVERSITY)

YENEPOYA INSTITUTE OF ARTS, SCIENCE, COMMERCE AND MANAGEMENT A CONSTITUENT UNIT OF YENEPOYA (DEEMED TO BE UNIVERSITY) BALMATTA, MANGALORE

Automated Vulnerability Assesment Tool

PROJECT SYNOPSIS

Automated VulnerabilityTesting Tool

BACHELOR OF SCIENCE

Cyber Forensics, Data analytics & Cyber Security

SUBMITTED BY :

Abhijith A Nair 21633

Abhinav K S 22064 (Team Leader)

Gibin G 21650

Manuraj Menon 22062

Rahul Kuruvath 23058

GUIDED BY

Mr. Shashank

PROJECT: InfoGather Pro - Cybersecurity Information Tool

Synopsis

InfoGather Pro is a web-based cybersecurity reconnaissance tool developed to support ethical hackers and security analysts in performing automated intelligence gathering. The tool focuses on collecting publicly available information about internet-facing domains to streamline the reconnaissance phase of security assessments.

The primary goal of InfoGather Pro is to reduce the manual effort involved in gathering preliminary data about a web target by automating key reconnaissance tasks. It performs the following operations:

- **DNS Records**: Gathers records such as A, AAAA, MX, NS, TXT, CNAME, SOA, and PTR. It also parses TXT records for SPF and DMARC configurations, and performs reverse DNS lookups to retrieve hostname mappings for discovered IPs.
- **WHOIS Information**: Extracts domain registration metadata including registrar, creation and expiration dates, name servers, contact emails, and country of registration.
- **Subdomain Enumeration**: Utilizes Certificate Transparency Logs (via crt.sh) to discover subdomains that are associated with SSL/TLS certificates. This helps reveal hidden infrastructure not listed on the main site.
- **SMTP Diagnostics**: Identifies mail servers using MX records and attempts SMTP connections to assess server response, availability, and basic functionality (such as banner capture and HELO verification).
- **Blacklist Status**: Checks IP addresses against DNS-based blacklists (DNSBLs) to determine if they are flagged for spam, abuse, or malware hosting. This is crucial in evaluating a domain’s reputation.

TABLE OF CONTENTS

1. INTRODUCTION

In today's digital landscape, the frequency and sophistication of cyber threats have significantly increased. Organizations face continuous risks from adversaries who exploit publicly available information to discover vulnerabilities. In this context, the role of ethical hackers and cybersecurity analysts has become more crucial than ever. These professionals rely on reconnaissance tools to gather intelligence that aids in identifying security flaws early in the cybersecurity assessment process.

InfoGather Pro is a comprehensive, full-stack cybersecurity reconnaissance tool developed to automate the process of collecting intelligence about internet-facing web assets. The tool is designed to enhance the efficiency and accuracy of the initial information-gathering phase, which is a critical step in penetration testing, threat modeling, and digital forensics.

At its core, InfoGather Pro performs a range of automated tasks that traditionally require the use of multiple independent tools. These include:

- **Metadata Extraction**: Captures page titles, meta descriptions, response headers, server banners, and basic technology fingerprinting.

- **DNS Analysis**: Retrieves A, AAAA, MX, NS, TXT, CNAME, SOA, and PTR records; parses SPF and DMARC policies from TXT records; and performs reverse DNS lookups.

- **WHOIS Lookup**: Provides domain registration details such as registrar, creation and expiration dates, name servers, administrative contacts, and registrant location.

- **Subdomain Enumeration**: Uses data from Certificate Transparency Logs via crt.sh to identify subdomains associated with SSL certificates, which can reveal additional attack surfaces.

- **SMTP Diagnostics**: Connects to mail servers listed in MX records, verifies connectivity and banner responses, and checks for misconfigurations through HELO communication.

- **Blacklist Status Check**: Cross-checks discovered IP addresses against popular DNS-based blacklists (DNSBLs) to identify flagged or suspicious hosts.

  The **backend** is powered by the Python Flask framework (app.py), which manages the scanning logic, handles API routes, and executes multi-threaded scans to ensure non-blocking performance. It uses specialized libraries such as dnspython, python-whois, smtplib, socket, and requests to interact with external servers and services.

  The **frontend** is built with HTML, CSS (using Bootstrap 5), and JavaScript. It offers a responsive, dark-themed interface that supports domain input validation, loading animations, and accordion-style result sections. The interface is intuitive, allowing users to quickly visualize scan results categorized under metadata, DNS, WHOIS, subdomains, SMTP diagnostics, and blacklist checks

  In summary, InfoGather Pro addresses a fundamental need in the cybersecurity field: reliable, efficient, and automated reconnaissance. Its extensible architecture and real-time performance make it an ideal tool for ethical hackers, penetration testers, security analysts, and students in cybersecurity programs. Through automation and thoughtful interface design, it bridges the gap between manual intelligence gathering and scalable, modern threat assessment workflows.

2. OBJECTIVES

The development of **InfoGather Pro** is guided by key objectives aimed at improving the efficiency, accuracy, and usability of the reconnaissance phase in cybersecurity assessments. Each objective has been designed to align with the practical needs of cybersecurity professionals and to support future scalability.

1. Automating the reconnaissance phase of penetration testing

One of the primary goals of InfoGather Pro is to minimize the manual workload involved in early-stage security assessments. Traditional reconnaissance involves using several tools independently to gather DNS data, WHOIS records, subdomains, and more. InfoGather Pro brings all these functionalities under one automated process, drastically reducing the time and effort required while maintaining accuracy and depth in data collection.

2. Providing a centralized and simplified interface for intelligence gathering

Instead of relying on separate utilities or command-line scripts, InfoGather Pro offers a unified web interface where users can initiate scans, view categorized results, and interact with the data easily. This centralization ensures that security analysts can manage tasks more efficiently and with less technical overhead, especially useful for teams that require streamlined operations.

3. Ensuring structured data presentation and export options

The tool structures all collected information into clear, categorized sections—such as metadata, DNS, WHOIS, subdomains, SMTP diagnostics, and blacklist checks. These results are not only easy to interpret in the browser interface but can also be exported in JSON format. This feature allows users to retain a copy of their findings, integrate them into reports, or feed the results into other analysis pipelines.

4. Designing an extendable and modular architecture

InfoGather Pro has been built with scalability in mind. Its modular backend architecture makes it easy to add new scanning features, integrate third-party APIs, or connect with other cybersecurity platforms. This forward-thinking design ensures that the tool can evolve with the changing needs of users and remain compatible with emerging security tools and frameworks.

3.METHODOLOGY/DEVELOPMENT PLAN

Phase 1: Requirement Analysis & Tool Selection

The project began with a thorough requirement analysis, where various reconnaissance techniques and cybersecurity needs were examined. Based on this, tools and libraries like dnspython for DNS lookups, whois for domain registration details, and sublist3r for subdomain enumeration were selected.

Phase 2: Backend Development with Flask

The backend logic was implemented using the Flask framework. This phase involved setting up API endpoints for each scan type, such as DNS, WHOIS, and SMTP. Functions were developed in a modular manner to make the codebase maintainable and extendable. JSON was used as the primary data exchange format.

Phase 3: Frontend Design and Integration

The frontend was designed using HTML and styled using Bootstrap 5 to ensure responsiveness and ease of use. Dark mode support was added for user preference. JavaScript handled dynamic updates and user interactions, displaying scan results in real time after each query was made to the backend APIs.

Phase 4: Testing and Debugging

Multiple test cases were conducted to validate the tool across a variety of domain types. Error handling mechanisms were implemented to manage edge cases, such as unreachable domains, API failures, or timeouts during data fetching operations.

Phase 5: Documentation and Packaging

After functionality was verified, comprehensive documentation was prepared. Features like scan result export in JSON format and final project packaging were added. This ensures usability for real-world application and integration with other tools or reports.

4. TOOLS AND TECHNOLOGIES USED

The development of **InfoGather Pro** relies on a combination of software and hardware resources that together support its full-stack functionality. The tool is designed to be lightweight, portable, and accessible on standard computing setups.

Software Requirements

• Python 3.x
Python serves as the primary programming language for backend development. It is known for its readability, wide range of libraries, and robust community support. Python's versatility enables rapid development of networking tools and integration with APIs and system-level resources.

• Flask
Flask is a lightweight Python web framework used to build the backend API of InfoGather Pro. It handles HTTP request routing, integrates smoothly with templates, and provides a clean structure for managing multiple scanning functions. Flask was chosen for its simplicity, scalability, and ease of use in building RESTful services.

• HTML, CSS (Bootstrap), and JavaScript
These technologies form the core of the frontend.

- **HTML** structures the content of the web interface.
- **CSS (with Bootstrap 5)** provides responsive design, dark mode styling, and consistent UI elements across devices.
- **JavaScript** handles client-side logic such as input validation, interactive elements, API calls, and dynamic result rendering.

• Libraries and Modules

- **dnspython**: Performs DNS record lookups, including advanced queries like reverse DNS (PTR), SPF, and DMARC records.
- **whois**: Retrieves domain ownership details, registration dates, and status information from WHOIS databases.
- **smtplib**: Used to connect to and interact with mail servers (via SMTP) for banner capture and availability testing.
- **requests**: Enables HTTP requests to retrieve metadata and interact with external services like crt.sh.
- **beautifulsoup4**: Parses HTML content to extract metadata like page title and description from target websites.

Hardware Requirements

• Standard PC or Laptop (Minimum 4GB RAM)
The tool is designed to run efficiently on any modern personal computer or laptop with at least 4GB of RAM. It does not require a high-end system or specialized hardware, making it accessible to students, analysts, and ethical hackers using everyday devices.

• Stable Internet Connection
A reliable internet connection is essential for the tool to function properly. Most of the scans—such as DNS lookups, WHOIS queries, subdomain enumeration, and blacklist checks—depend on real-time communication with external servers and APIs. Network timeouts or disconnections may affect scan accuracy and completeness. 5.

5.SYSTEM ARCHITECTURE

- **Presentation Layer (Frontend):**
    - **Components:** index.html, style.css (Bootstrap 5 enhanced), script.js.
    - **Functionality:** This layer is responsible for the user interface (UI) and user experience (UX). It provides an intuitive, responsive, and dark-themed web interface where users can input a target domain or URL. Key features include:
        - Input validation for the target.
        - Dynamic loading animations during scans.
        - Accordion-style, categorized display of scan results (Metadata, DNS, WHOIS, Subdomains, SMTP Diagnostics, Blacklist Status).
        - Client-side logic for interacting with the backend API.
        - Functionality to download scan results in JSON format.
    - **Technologies:** HTML5, CSS3, Bootstrap 5, JavaScript (ES6+).
- **Application Layer (Backend):**
    - **Component:** app.py (Flask application).
    - **Functionality:** This layer serves as the core logic engine of the application. It handles incoming API requests from the frontend, orchestrates the various scanning tasks, and processes the data. Its responsibilities include:
        - Defining API endpoints (e.g., /scan).
        - Managing and executing reconnaissance modules for:
            - DNS record lookups (A, AAAA, MX, NS, TXT, CNAME, SOA, PTR, SPF/DMARC parsing).
            - WHOIS information extraction.
            - Subdomain enumeration via Certificate Transparency logs (crt.sh).
            - SMTP server diagnostics (connection, banner, HELO).

        - IP address blacklist checks (DNSBLs).
        - Website metadata extraction.
    - Utilizing multi-threading (e.g., ThreadPoolExecutor) for non-blocking execution of concurrent tasks like DNSBL checks.
    - Formatting and returning scan results as JSON to the frontend.
  - **Technologies:** Python 3.x, Flask.
  - **Key Libraries:** dnspython, python-whois, smtplib, socket, requests, BeautifulSoup4, concurrent.futures.

- **Data Access & External Services Layer:**
  - **Functionality:** This conceptual layer represents the interaction of the backend with external data sources and services. It's not a separate code component but rather how the backend fetches information.
    - **DNS Resolvers:** Queries public or specified DNS servers for various record types.
    - **WHOIS Servers:** Connects to TLD-specific WHOIS servers to retrieve registration data.
    - **crt.sh API:** Queries the Certificate Transparency log search engine for subdomains.
    - **Target Web Servers:** Makes HTTP/S requests to fetch metadata and HTML content.
    - **Mail Servers (SMTP):** Connects to mail servers identified via MX records for diagnostics.
    - **DNSBL Servers:** Performs DNS lookups against various blacklist servers.
  - **Communication Protocol:** Primarily DNS, WHOIS protocol, HTTP/S, SMTP.

- **API Communication:**
  - The **Frontend** (Presentation Layer) communicates with the **Backend** (Application Layer) via asynchronous HTTP POST requests (using the fetch API in JavaScript) to the /scan endpoint.
  - Data is exchanged in JSON format, with the frontend sending the target URL and the backend returning a comprehensive JSON object containing all scan results

6.EXPECTED OUTCOME

Upon successful completion, InfoGather Pro will serve as a robust, user-friendly, and efficient cybersecurity reconnaissance tool. The key outcomes anticipated are:

1. Streamlined Reconnaissance: The tool will significantly reduce the manual effort and time involved in the initial information-gathering phase of security assessments by automating the collection of diverse data points (DNS, WHOIS, subdomains, SMTP, blacklist status, metadata) into a single, unified process.

2. Centralized & Intuitive Web Interface: Users will benefit from a clear, interactive, and dark-themed web interface for initiating scans and viewing well-categorized results. This eliminates the need to use multiple disparate command-line tools, making intelligence gathering more accessible, especially for teams and individuals preferring a GUI.

3. Structured Data Presentation & Export: All collected information will be presented in an organized, easily digestible format within the browser. The ability to export the complete scan results as a JSON file will facilitate:

    - Offline analysis and archiving.
    - Easy integration into security reports.
    - Feeding data into other security tools or analysis pipelines for further investigation.

4. Enhanced Situational Awareness: By providing a comprehensive overview of a target's internet-facing assets and potential vulnerabilities (e.g., misconfigured SMTP, blacklisted IPs), the tool will empower ethical hackers, penetration testers, and security analysts to make more informed decisions during security assessments and threat modeling.

5. Foundation for Future Development: The modular architecture of InfoGather Pro will establish a solid base for future enhancements. This includes the potential to:

    - Integrate additional scanning modules (e.g., port scanning, vulnerability feeds).
    - Support more third-party APIs for richer data.
    - Develop advanced reporting features.
    - Introduce user accounts and scan history.

6. Educational Value: The tool will serve as a practical learning resource for students in cybersecurity programs, allowing them to understand and experiment with various reconnaissance techniques in a controlled and ethical manner.

7.REFERENCES

- Official Documentation: dnspython, Flask, Python-WHOIS
- OWASP Reconnaissance Guide
- crt.sh (Certificate Transparency Log)
- Sublist3r GitHub Repository
- BeautifulSoup4 Documentation