Custom Reconnaissance Tool Development

ITSolera internship

AUTHOR: MAHMOUD SALAH MAHMOUD

DATE: 13TH JUNE 2025

TOOL NAME: ITSOLERA RECON SCANNER

INTERNSHIP PROGRAM: ITSOLERA CYBER DEPARTMENT – SUMMER INTERNSHIP 2025

Table of Contents

Executive Summary.....3

Technical Details 3

Key Features 3

- Modular CLI Interface 3
- Execution Method 3
- Active Recon Modules 3
- Passive Recon Modules.....3

Output Summary 3

Screenshots & Evidence.....4

Conclusion.....6

Executive Summary

This report presents the design and implementation of **ITSOLERA Recon Scanner**, a Python-based modular reconnaissance tool created as part of the ITSOLERA Cyber Department's Summer Internship 2025.

The tool automates information gathering during the reconnaissance phase of penetration testing engagements by integrating passive and active recon modules. It is designed to be CLI-driven, lightweight, and extendable — allowing security analysts and red teamers to collect data on target domains quickly and efficiently.

All results are printed in the terminal, with subdomain results stored in a dedicated `.txt` file for documentation or reuse in further exploitation steps.

This is the GitHub link of the project: https://github.com/Mahmoud-Sa1ah/Mahmoud-Sa1ah-Reconnaissance-Automated-Tool/

Key Features

◆ Modular Command-Line Interface

Each module can be triggered individually using specific CLI flags:

◆ Execution Method

The tool was executed using specific CLI flags to activate modules (through the terminal).

```
python main.py example.com --whois --dns --subdomains --ports
```

◆ Passive Reconnaissance

- **WHOIS Lookup**: Extracts domain registration and expiry information.
- **DNS Enumeration**: Retrieves A, MX, TXT, and NS records.
- **Subdomain Enumeration**: Uses `crt.sh` and `HackerTarget` APIs to gather subdomains.

◆ Active Reconnaissance

- **Port Scanning**: Uses `nmap` (via wrapper) to identify open TCP ports.
- **Directory Enumeration**: Probes web root for common file/directory names.
- **Basic Vulnerability Scan**: Identifies exposed sensitive files and misconfigurations.

Technical Overview

- **Programming Language**: Python 3.12
- **Framework**: Custom CLI using `argparse`
- **Development Environment**: Windows 11 using PyCharm
- **Dependencies**:
  - `requests`, `whois`, `dnspython`, `argparse`, `nmap`, `yaml`, `BeautifulSoup`

Directory Structure

```
ITSOLERA/
├── main.py                      # Entry point with CLI argument handling
├── providers/
│   ├── __init__.py              # Package initializer
│   ├── whois_lookup.py          # WHOIS functionality
│   ├── dns_enum.py              # DNS enumeration
│   ├── crtsh.py                 # Certificate transparency log analysis
│   ├── hackertarget.py          # Integration with HackerTarget API
│   ├── active_recon.py          # Active scanning logic (e.g., live host)
│   ├── port_scan.py             # Port scanning implementation
│   ├── dir_enum.py              # Directory brute-force enumeration
│   └── vul_scan.py              # Basic vulnerability heuristics
├── config/
│ └── sources.yaml
```

Output: appears in the terminal and this detailed pdf report for the selected target.

Output Summary

- **Target Domain**: mars.com

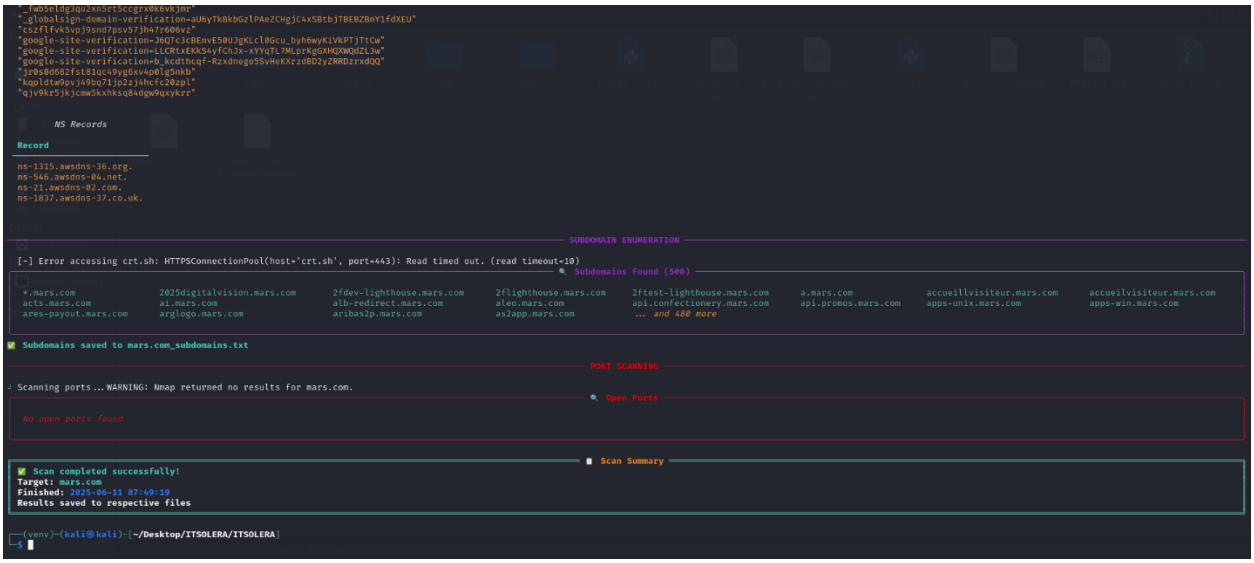
  I got this target from HackerOne platform, a real-world bug bounty platform.

- **WHOIS Lookup**: Successfully retrieved registrar, creation date, expiry date.
- **DNS Records**: A, MX, TXT, NS found.
- **Subdomains**:
  Saved to file: `mars.com_subdomains.txt`
  Example discovered:

  ```
  a.mars.com
  api.confectionery.mars.com
  arglogs.mars.com
  as24op.mars.com
  b-redirect.mars.com
  acomelivisiteur.mars.com
  2flighthouse.mars.com
  ```

- **Port Scanning**:
  - No Open Port Found
- **Directory Enumeration**:
  - No significant directories or files found
- **Vulnerability Scan**:
  - No vulnerabilities found

Screenshots & Evidence



Conclusion

The **ITSOLERA Recon Scanner** successfully achieves its goal of providing a customizable, scriptable reconnaissance tool for real-world red team operations.

By modularizing each recon step and enabling flexibility through CLI flags, the tool allows ethical hackers to tailor recon workflows for different targets. Although output is primarily displayed in the terminal, subdomains are saved in a `.txt` file, and logging support can be expanded in future versions.

The project was completed on time, meets all Task requirements, and includes clear structure, modular coding standards, and potential for Docker deployment in the next iteration.