Red Team Multi-Site Subdomain Recon Tool

Overview

The Red Team Multi-Site Subdomain Recon Tool is a professional-grade reconnaissance utility designed for red team engagements. It enables security professionals to enumerate subdomains across multiple target websites, classify web pages (e.g., login, admin, API), and capture screenshots for rapid analysis. Built with a focus on efficiency and stealth, the tool combines active and passive reconnaissance techniques to uncover hidden subdomains and potential attack surfaces.

Features

Multi-Site Subdomain Enumeration: Actively brute-force subdomains using a customizable wordlist and passively gather subdomains via public sources (e.g., crt.sh).
Page Classification: Automatically categorize web pages (e.g., login, admin, API) using a keyword-based system.
Screenshot Capture: Capture screenshots of live subdomains using a headless Chrome browser, organized by category.
Professional GUI: A sleek, Streamlit-based interface with customizable themes (Dark Gradient, Cyber Neon) and an eye-catching design.
Red Team Focus: Prioritize sensitive subdomains (e.g., staging, internal, backup) for identifying misconfigurations.
Extensible: Customizable wordlists for subdomains and keywords, allowing tailored reconnaissance.

Project Structure

```
subdomain_recon_tool/
├── main.py # Streamlit app with GUI
├── recon.py # Subdomain enumeration and screenshot logic
├── utils.py # Helper functions (classification, logo, keywords)
├── wordlists/ # Wordlist folder
│ ├── subdomains.txt # Subdomain brute-force wordlist (100 entries)
│ └── keywords.txt # Keyword wordlist for classification
├── assets/ # Static assets
│ └── logo.png # Tool logo
├── screenshots/ # Output folder for screenshots
│ ├── login/ # Login page screenshots
│ ├── admin/ # Admin page screenshots
│ ├── api/ # API endpoint screenshots
│ └── other/ # Other page screenshots
└── requirements.txt # Dependencies
```

Installation

Clone the Repository (if applicable):

```
git clone
cd subdomain_recon_tool
```

Set Up a Virtual Environment (recommended):

```
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate
```

Install Dependencies:Ensure you have Python 3.8+ installed, then run:
pip install -r requirements.txt

Dependencies include:

streamlit==1.29.0
aiohttp==3.9.1
dnspython==2.4.2
selenium==4.16.0
webdriver-manager==4.0.1

Install Chrome and ChromeDriver:

Ensure Google Chrome is installed.
The webdriver-manager package automatically handles ChromeDriver, but if issues arise, manually download ChromeDriver from chromedriver.chromium.org and add it to your PATH.

Add a Logo (optional):

Place a logo.png file in the assets/ directory. A default placeholder is used if missing.

Usage

Run the Tool:
streamlit run main.py

This will launch the GUI in your default browser (e.g., http://localhost:8501).

Configure the Scan:

In the sidebar, enter target domains (one per line, e.g., example.com, test.com).
Specify the output directory for screenshots (default: screenshots).
Provide paths to the subdomain and keyword wordlists (default: wordlists/subdomains.txt, wordlists/keywords.txt).
Select a theme (Dark Gradient or Cyber Neon).

Launch the Scan:

Click "Launch Recon" to start the scan.
The tool will enumerate subdomains, classify pages, and capture screenshots.

View Results:

Results are displayed in expandable sections, showing the subdomain, URL, category, screenshot, and passive source.
Screenshots are saved in the screenshots/ directory, organized by category (e.g., screenshots/login/).

Example
Input:

Domains: example.com, test.com
Output Directory: screenshots

Wordlist: wordlists/subdomains.txt

Output:

Discovered subdomains (e.g., admin.example.com, api.test.com).
Screenshots saved (e.g., screenshots/login/example.com_20250604_1037.png).
GUI displays results with categorized screenshots.

Customization

Subdomain Wordlist: Edit wordlists/subdomains.txt to add more subdomains for brute-forcing.
Keyword Wordlist: Modify wordlists/keywords.txt to define new categories or keywords for page classification.
Logo: Replace assets/logo.png with your custom logo in PNG format.

Requirements

Python 3.8 or higher
Google Chrome (for screenshot capture)
Internet connection (for passive recon and HTTP requests)

Notes for Red Team Professionals

Stealth: Passive recon via crt.sh minimizes active scanning footprints.
Targeted Recon: The tool prioritizes sensitive subdomains (e.g., internal, backup) that are often misconfigured.
Extensibility: Add custom subdomains and keywords to focus on specific attack surfaces.

Troubleshooting

Screenshots Not Capturing:
Verify Chrome and ChromeDriver compatibility.
Ensure internet connectivity.

GUI Not Loading:
Check Streamlit installation (streamlit --version).
Run the tool in a virtual environment to avoid dependency conflicts.

Logo Missing:
Ensure logo.png is in the assets/ directory, or the placeholder will be used.

License
This tool is intended for authorized security testing and red team engagements only. Use responsibly and in compliance with applicable laws and permissions.
Author
Developed by a cybersecurity red team professional for advanced reconnaissance tasks. For issues or contributions, contact the project maintainer.

Last Updated: June 04, 2025