Website Scanner Tool Documentation

Table of Contents

Introduction

The Website Scanner Tool is a Python-based utility that allows you to perform a variety of security scans on a target website. It integrates several tools and checks into a single script, including open port scanning, SSL/TLS security checks, HTTP header fetching, and basic vulnerability scanning like XSS and SQL Injection detection.

Features

- **Port Scanning**: Uses nmap to detect open ports on the target server.
- **SSL/TLS Security Check**: Uses sslscan to evaluate the SSL/TLS configuration of the target website.
- **HTTP Header Analysis**: Fetches and displays HTTP headers using the requests library.
- **Vulnerability Scanning**: Basic checks for XSS and SQL Injection vulnerabilities.

Prerequisites

Before you can run the Website Scanner Tool, make sure your environment meets the following requirements:

- **Python 3.x** installed.
- **nmap**: A network scanning tool.
- **sslscan**: A tool to check SSL/TLS security.
- **Python libraries**: Installable via pip (see installation instructions below).

Installation

Step 1: Clone the Repository

Clone the repository to your local machine:

```
git clone <repository_url>
```

```
cd website_scanner
```

Step 2: Install Python Dependencies

```
pip install -r requirements.txt
```

1. **Ensure Required Tools are Installed**

Make sure nmap and sslscan are installed on your system. On Ubuntu/Debian, you can install them using:

```
sudo apt-get install nmap sslscan
```

For other systems, use the appropriate package manager or download the binaries from the official sites.

Usage

To run a scan on a target website, use the following command:

```
python3 website_scanner.py <url>
```

Replace <url> with the target website's URL, for example:

```
python3 website_scanner.py example.com
```

Output:

The tool will display the following information in sequence:

1. **IP Address**: The IP address of the target URL.
2. **Port Scan Results**: A list of open ports.
3. **SSL/TLS Security Report**: Details about the SSL/TLS configuration.
4. **HTTP Headers**: The headers returned by the target server.
5. **Vulnerability Scan Results**: Detection of potential XSS or SQL Injection vulnerabilities.

Error Handling and Troubleshooting

Common Issues and Fixes

1. **nmap or sslscan not found:**
    - Ensure that these tools are installed and accessible in your system's PATH.
2. **Python module not found:**
    - Ensure that you have installed all required Python dependencies using pip install -r requirements.txt.
    - Check the utils/ directory structure to ensure all files are in the correct place.
3. **Network-related issues:**
    - Make sure you have a stable internet connection.
    - Ensure the target URL is correct and accessible.
4. **Permission Denied Errors:**
    - Running network scanning tools may require elevated privileges. Try running the script with sudo (on Unix-based systems) if necessary.

Contributing

If you'd like to contribute to this project, please follow these steps:

1. Fork the repository.
2. Create a new branch (git checkout -b feature-branch).
3. Make your changes.
4. Commit your changes (git commit -m 'Add new feature').
5. Push to the branch (git push origin feature-branch).

6. Open a Pull Request.

Please ensure that your code adheres to the existing code style and passes any tests before submitting a PR.

License

This project is licensed under the MIT License. See the LICENSE file for more details.

This documentation provides an overview of the Website Scanner Tool, installation instructions, usage guidelines, a description of the file structure, and troubleshooting tips. It should give you everything you need to get started with the tool and contribute to its development.