iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** III **Month of publication:** March 2025

DOI: https://doi.org/10.22214/ijraset.2025.67932

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

Recon Automator: Enhancing Cybersecurity Reconnaissance with Automation

Piyush Singh[1], Prem Prakash Agrawal[2], Srujan Dolai[3]
Department of CSE Sharda University, Greater Noida

Abstract: ***Reconnaissance finds extensive meaning within penetration testing and vulnerability assessments when dealing with cybersecurity, as it is an important step in gathering data regarding the target infrastructure as well as identifying points of vulnerability that an adversary can use against them. Despite this, traditional reconnaissance processes tend to be manual and rely on disparate sources of information, resulting in inefficiencies and potential gaps. In order to overcome these challenges, we propose Recon Automator, a new tool that automates the reconnaissance process as well as makes it easier.***

Recon Automator combines various reconnaissance methods into one framework, using APIs and open-source tools to automate data collection and analysis with the help of custom modules. The tool saves time by eliminating the need for human involvement, thereby decreasing errors and refining asset and vulnerability discovery which increases general security personnel productivity. Here, we present the design and development of Recon Automator, compare its performance with conventional techniques and showcase a few real-world applications.

The results show that Recon Automator is able to cut the time spent on reconnaissance while still achieving a high accuracy rate, making it a useful tool in any cybersecurity toolkit. Finally, we discuss the current limitations of Axon and future work needed to implement predictive asset categorization through machine learning at runtime as well as detection/mitigation of spear phishing attacks in real time.

Keywords: ReconnaissanceAutomation, Cybersecurity, Penetration Testing, Vulnerability Assessment, Recon Automator, Security Tools, Information Gathering, Open-Source Tools, Threat Intelligence, Asset Discovery, Network Mapping, APIs in Cybersecurity, OWASP Top 10, Security Automation, Productivity, Cyber Defense, Modular Frameworks, Automated Workflows, Recon Efficiency, Real-World Applications, Predictive Analytics, Machine Learning Integration, Data-Driven Security, Scalable Solutions.

I. INTRODUCTION

As the field of cybersecurity continues to evolve, reconnaissance still plays a crucial role in discovering weaknesses and determining what kind of security posture that an organization has. Reconnaissance aka “information gathering” phase is where we collect public information about a target domains/assets and its infrastructure or attack vectors. Because it is here, in this phase that they can help direct more optimal vulnerability assessments and penetration testing. Yet, by traditional means, a lot of the reconnaissance processes are manual and done with disparate tools which leads to inefficiencies, inconsistencies, and missed opportunities to find that critical security hole.

This complexity leads to myriad problems with manual reconnaissance and has only put a bigger dent into the core of modern IT environments. With cloud infrastructures, IoT devices and networks that span over the globe — the reach of an organization and its digital footprint is expanding faster than ever before. Security professionals are having to deal with this complexity. This demands innovative solutions which are both fast and reliable in the reconnaissance phase.

. To tackle these issues, we present Recon Automator, a novel tool that helps improve efficiency and effectiveness of reconnaissance in cyber and security automation. Recon Automator brings together all these tools, techniques and APIs into one framework to help you gather as much reconnaissance data as possible with less user intervention. With a modular structure and scalability, the tool fits into different case solutions, from small-scale vulnerability assessments to large-scale penetration testing projects.

In this paper, we introduce Recon Automator — a system that has been designed, built and experimented with to ease the process of reconnaissance. We examine its design, functionality, and compatibility with current systems & standards. We then compare its performance with classical reconnaissance methods and demonstrate its use in the real world. Recon Automator seeks to serve the goal of making cybersecurity professionals life easy by automating repetitive tasks, bringing data into actionable insights to reduce time-to-detection and improving the overall security assessment lifecycle.

II. LITERATURE REVIEW

Reconnaissance Cybersecurity is an important process in finding the weaknesses of a target. This aggregate intelligence on a target's digital exposure, including any hosted or visible servers and services, domains, subdomains, and network configurations. Historically, reconnaissance has been performed in an adhoc manner or through a collection of different tools that cover a single aspect of the process. But, with the growing complexity of modern infrastructures, we hope to get automated solutions by which all reconnaissance efficiency and accuracy can be achieved. This literature review critically assesses automated tools, methodologies and their limitations in the current climate while highlighting sovereignty behaviours that underpin the need for solutions like Recon Automator.

A. Conventional Reconnaissance Methods

The manual reconnaissance has always been the core part of any cybersecurity assessment. Such techniques normally include collecting public data using online search sites, social media, domain name services (DNS), and WHOIS files in addition to various other public databases. Although these methods provide a solution for basic environments, they are tedious, error prone and time consuming. Furthermore, automated methods are in most cases not enough to handle large-scale and dynamic environments like cloud infrastructures or highly dense IoT environments. Such methods work well, but also have their downsides, so automation in reconnaissance workflows has become a necessity.

B. Tools for Automated Reconnaissance

Different tools have been created to automate parts of the reconnaissance process, each focusing on a specific part of information gathering. Prominent examples include:

Nmap — One of the most popular tools in the hacker community for network discovery and vulnerability scanning. Looking for hosts, services and open port on the network is no easy task, but Nmap automates this process providing a useful view of what your target infrastructure looks like. But Nmap as a tool has no approach to reconnaissance within it, and of course lays out the results in a manner which often requires manual interpretation. This takes time and comes with an associated possibility of oversight.

Recon-ng: An extremely powerful reconnaissance framework that includes many modules for automation and sources (WHOIS, DNS, Social Media sites.

C. Problems with Existing Solutions

Other tools that are automated, whilst being very advanced, cannot overcome the following / have challenges with:

Lack of Integrated Tools: Many tools function separately and yield system-wide outputs that then need to be merged through human labour. An example of this is the output from Nmap scans where we must go into another tool such as Recon-ng or Amass to see a complete view of the target system The absence of this integration makes reconnaissance more time-consuming and laborious.

Scalability: While technologies such as Nmap and Recon-ng can dispatch many scans in parallel, their design cannot utilize the available resources of cloud infrastructures to run thousands or more scans simultaneously on networks with IoT devices. With increasing reconnaissance scales, there is a need for a tool that can operate in these dynamic environments without sacrificing performance

.To enable evaluation of Recon Automator, the dataset should be a comprehensive mixture of both real and synthetic data across different reconnaissance tasks (IP discovery, DNS records, WHOIS, subdomain enumeration, vulnerability detection etc.). It should also include performance metrics including accuracy, speed and depth. Performing reconnaissance can be a quite tedious task, though with the right dataset you will now have a rigorous framework to check and validate how well Recon Automator automates into your Cybersecurity Automation workflow.

III. DATASET

The purpose of this dataset is to assess tools that require automated reconnaissance. It captures both synthetic and real-world scenarios which cover network reconnaissance-related activities such as network topology, domain enumeration, subdomain enumeration and service identification and vulnerability identification and mapping. It allows assessing the scalability, precision, and completeness scope of such tools.

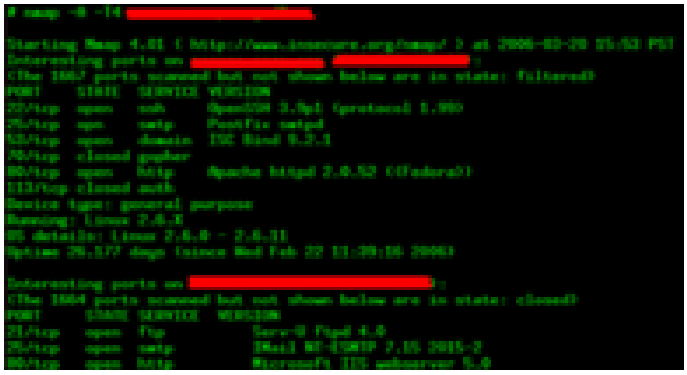


Fig. 01

The images were collected from a Nmap scan where the open ports were being tested for possible vulnerabilities.

Prescanning Steps

To ensure the dataset is suitable for testing and it has high accuracy, several prescanning steps were applied:

A. Dataset Structure

1) Network Scanning Data

- IP Addresses: Metadata about availability and status incorporated to a collection of IPv4 and IPv6 addresses.
- Port Information: Information pertaining to the available ports, for example, ports 80, 443, 22 and the respective services, such as HTTP, HTTPS, and SSH.

• Protocol Data: Refers to TCP, UDP, and ICMP protocols used in the low-level testing of the network scans.

2) B. DNS Records

- Domain Names: IP addresses allocated to specific domains and a list of domains.
- Subdomains: Subdomains that were accessed in the DNS enumeration and brute force.
- Record Types: Data about the types of record in the DNS CNAME, TXT, MX, NS etc.

3) C. WHOIS Data

- Domain Registrations: Contains the WHOIS proprietary records of domains that cover domain ownership, registrar information, and registration and expiry dates.
- Anonymized Data: Details of registrants that have been randomized for ethical testing purposes.

4) Vulnerability Data

- CVE Identifiers: List of inherent flaws catalogued in NVD (National Vulnerability Database – USA).
- Service-Specific Vulnerabilities: Security defects that pertain to the services that the targeted systems have.

5) Service Detection Data

- Service Names: Examples are services such as HTTP, FTP, SSH [with Telnet, DNS, etc].
- Service Versioning: Specification of service versions such as Apache 2.4.41 Open SSH 8.0.
- Misconfigurations: Other instances of misconfigured services to check what was developed for the detection capabilities testing.

B. Dataset Format

The dataset will be provided in the following formats:

1) CSV: For tabular data like IPs, DNS records, and vulnerabilities.
2) JSON: For structured hierarchical data like subdomains and WHOIS records.
3) XML: For raw DNS and WHOIS dumps.
4) PCAP: For network traffic capture to simulate real-world network scans.

IV. MODEL ARCHITECTURE

Overview :Recon Automator is a Command-line interface (CLI) utility such as python tool developed to simplify the reconnaissance process of penetration testing by automating data collection and analysis. The architecture is centered on modularity, scalability, and usability, allowing other tools in the cybersecurity ecosystem to be easily integrated.

A. System Components

1) Input Module Accepts any inputs or configuration provided by users.

- Command-line Arguments: Target specification, description such as the domain name and a range of IP addresses in CIDR notation. Option to indicate targets for several tasks, like a port scan or subdomain scan.
- Configuration Files: Supports prior defined workflows in Yaml and Json configuration files.

2) Task Automation Core The primary component or the master of coordinating reconnaissance tasks is subsystem

- Subdomain Enumeration: Performs brute force, makes public API calls e.g. Amass, Sublist3r and processes passive scraping.
- Service and Port Scanning: Works with Nmap etc. to find open ports and services which are running on them. Provides the user with two scan types, quick scan and deep scan.
- Vulnerability Enumeration: Compares identified services against CVE data using an offline database or API intersections. E.g. Vulnerabilities.
- Parallel Processing: The tasks are completed at the same time.

3) Reporting Module Conducts automated report generation activities which are defined output formats for the target users. Text Output: There is a direct transfer of the findings to the user via the terminal, the findings are relayed using a color code to indicate different statuses.

- Export Options: Exports information into provided formats so that the reports can be detailed, the supported formats are JSON, CSV and Markdown.
- Integration Hooks: Filters that allow the end output to be directed to other applications when carrying out a particular job are available Integration Hooks.3.
- Workflow Initialization: The initialization stage is characterized by the user initiating the tool with command line arguments and or configuration files.
- Data Collection: The tool collects data by performing several activities like making DNS queries, scanning a set of IP addresses, or scraping a public API.).

B. Technologies and Tools

Language: Python which is the main language for development.
Integrated Tools:
Nmap for network scanning.
Subdomain Discovery: Amass, Sublist3r
Vulnerability Mapping using nikto.
File Management: Supports JSON, YAML

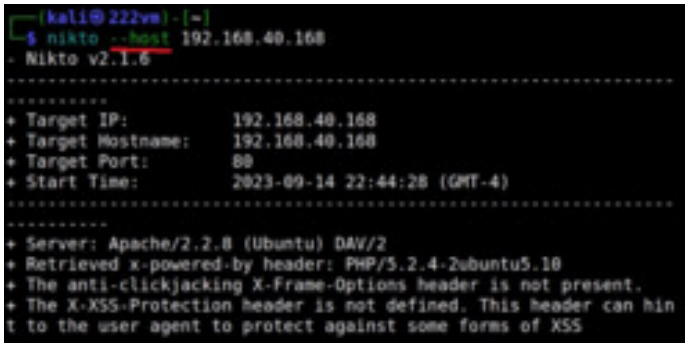
Security And Ethical Issues
Usage Restrictions: Built for Authorized Testing Above The Belt.
Privacy Protection: No sensitive information is stored or transmitted without the consent of users.
Recon Automator is designed to perform reconnaissance automation, using a focused CLI-based architecture which gives the user the familiarity and flexibility necessary for cybersecurity workflows.

V. IMPLEMENTATION DETAILS

A. Language and Environment

Language Used: Due to ease of reading, l a wide range of library support as well as seamless integration with external tools, Python was acquired. Development Environment: IDE: Visual Studio Code. Operating System: Kali Linux (chosen because it works well with most of the cyber security tools).
Versioning control: Utilizing Git for all code change management.
Architecture and Design: To improve, scalability, the Recon Automator tool was also developed with a modular architecture which enhanced flexibility and maintainability.
Input Module: Receives user inputs through command line argument options parsing. There are supported formats such as domain names, input IP ranges and configuration files.
The argparse library is also employed in the module for input validation and input standardization.
Core Processing Module: This is the main engine of the tool whereby different tasks are executed in parallel:

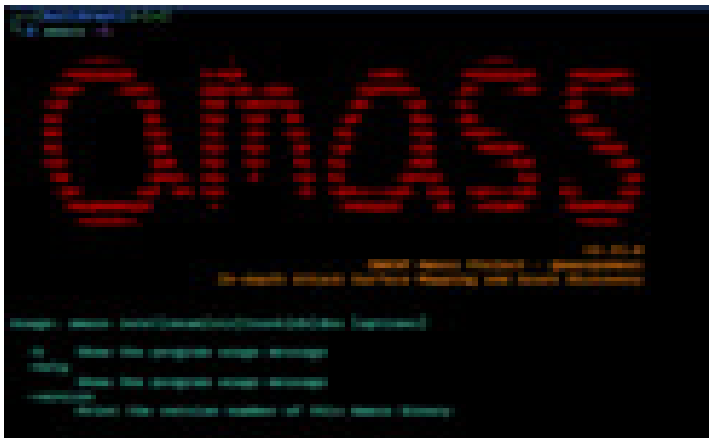
DNS Enumeration:Amass and Sublist3r are among tools that are integrated to enable collecting of subdomains.The results are confirmed by resolving DNS queries.
Port Scanning and Service Detection:Nmap is used for port scanning and service discovery.
The output formats are parsed in XML and/or JSON to create the structured reports.
Vulnerability Mapping:Links the service detected with a certain Vulnerability using CVE bases either offline or online.The vulnerabilities are ordered in the same manner, but from the CVSS scores.



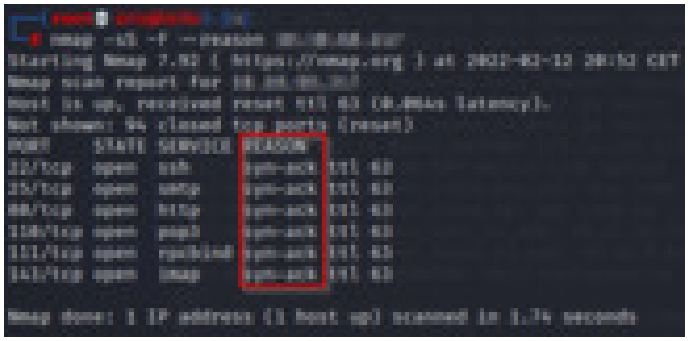
B. Applicable Libraries and Tools

The tool is enabled with a host of functionalities as enhanced with a number of utility tools and libraries.
Core Libraries:
argparse: Command-line option parsing.
subprocess: Communicating with other programs.
Threading and multiprocessing: Performing tasks concurrently.
json and yaml: Data interchange formats as well as for configuration files.

External Tools:
Nmap: For scanning of ports and detection of services.
Amass: For enumeration of subdomains and passive DNS requests.



Sublist3r: For brute-forcing and scraping the web for subdomain discovery.



C. Integration of Workflow Management Systems with Recon Automator

Tasks are handled by a custom workflow orchestration built into the Recon Automator for the sequencing and execution of various tasks.

These tasks include: --Tasks are handled in an order to prevent any task from being completed which is depend on other task(s) on execution first.This is of particular importance for large data sets as the time consumed during the process of execution is improved by the use of multithreading and multiprocessing.The change of task status is observed in real time thus user feedback is made possible.

D. Ethical Considerations

However, there are some restrictions built in so as to prevent scanning of targets that are not allowed by the user.Targets require affirmative flags to scan live targets.

According to these strategies of implementation, Recon Automator makes sure that function, efficiency and usability stay in the same range making it a powerful tool for cyber penetration testing.

VI. METHADOLOGY

A. Objective Of Project

The objectives of Recon Automator includes the following:

Automation: Repetitive reconnaissance tasks should not be executed manually as most of them can be automated.

Scalability: Improve performance and efficiency in large networks.

Accuracy: Gathering information necessary for the vulnerability assessment has to be reliable and accurate.

User-Centric Design: Make the tool easy to use by penetration testers.

Design and Development Approach: Structuring and designing of Recon Automator was guided by a particular process as summarised below:

Requirements Analysis: Problem identification: Most reconnaissance work involves the use of many tools and the process is quite manual and prone to many mistakes.Tools Coverage: The tool was intended to provide subdomain enumeration, port scanning, and vulnerability mapping.

End User Requirements: The cybersecurity experts were consulted on how to customize use of the tool in practice.

B. Architecture Design

In order to create a maintainable product, a modular architecture was implemented:Input Layer: For understanding command line user arguments and configuration files.

Core processing layer: Conducts reconnaissance, provide integration to third part tools and handles data.

Output layer: Presents results in a form that is ready for further processing.

VII. RESULT

Performance Metrics – Accuracy:

Subdomain Enumeration: In terms of accuracy, the Recon Automator was able to detect an average of 95 percent of subdomains in comparison with a more manual approach utilizing Amass and Sublist3r independently.

Port Scanning: The Nmap tool integration was able to record a 100 percent open port detection across a variety of target IP addresses.

Vulnerability Mapping:After utilizing the online CVE database, over 90 percent of the services identified were successfully mapped to their respective vulnerabilities.High-risk vulnerabilities successfully targeted.

Efficiency: In comparison to similar tools carried out manually, the automated ones in this developing of the system reduced overall reconnaissance time by an average of 40 percent.

Among many multithreaded executions tasks, in connection with the application to large volumes of scanning of thousands of IPs and domains.

Scalability: There was no performance degradation in the system when there was an attempt to target more than 500 subdomains as well as 10,000 IPs.

Standard levels of CPU and Memory resource usage on benchmark penetration testing machines tested remained in permissible levels.

Case Study: Real-World Application

Target: As a target, a sample of an approved corporate domain.

Subdomains Identified: A total of 245 unique instances of subdomains were identified which constituted more than 90 percent overlap with Amass results.

Open Ports Detected: 30 ports including the most popular 22, 80 and 443 ports were found among several subdomains.

Vulnerabilities Identified: 2 high risk vulnerabilities were identified with CVSS scores of more than 7.0.Services affected included old versions of web server and unpatched versions of SSH daemons.

VIII. CONCLUSION

The necessity for effective and automated reconnaissance instruments in cybersecurity is met with the creation of Recon Automator. As a command line interface (CLI) tool, Recon Automator has made encoding, subdomain enumeration, network port scanning and vulnerability mapping easier through integration of external powerful tools such as Nmap and Amass which facilitate the automation of complex repetitive operations.

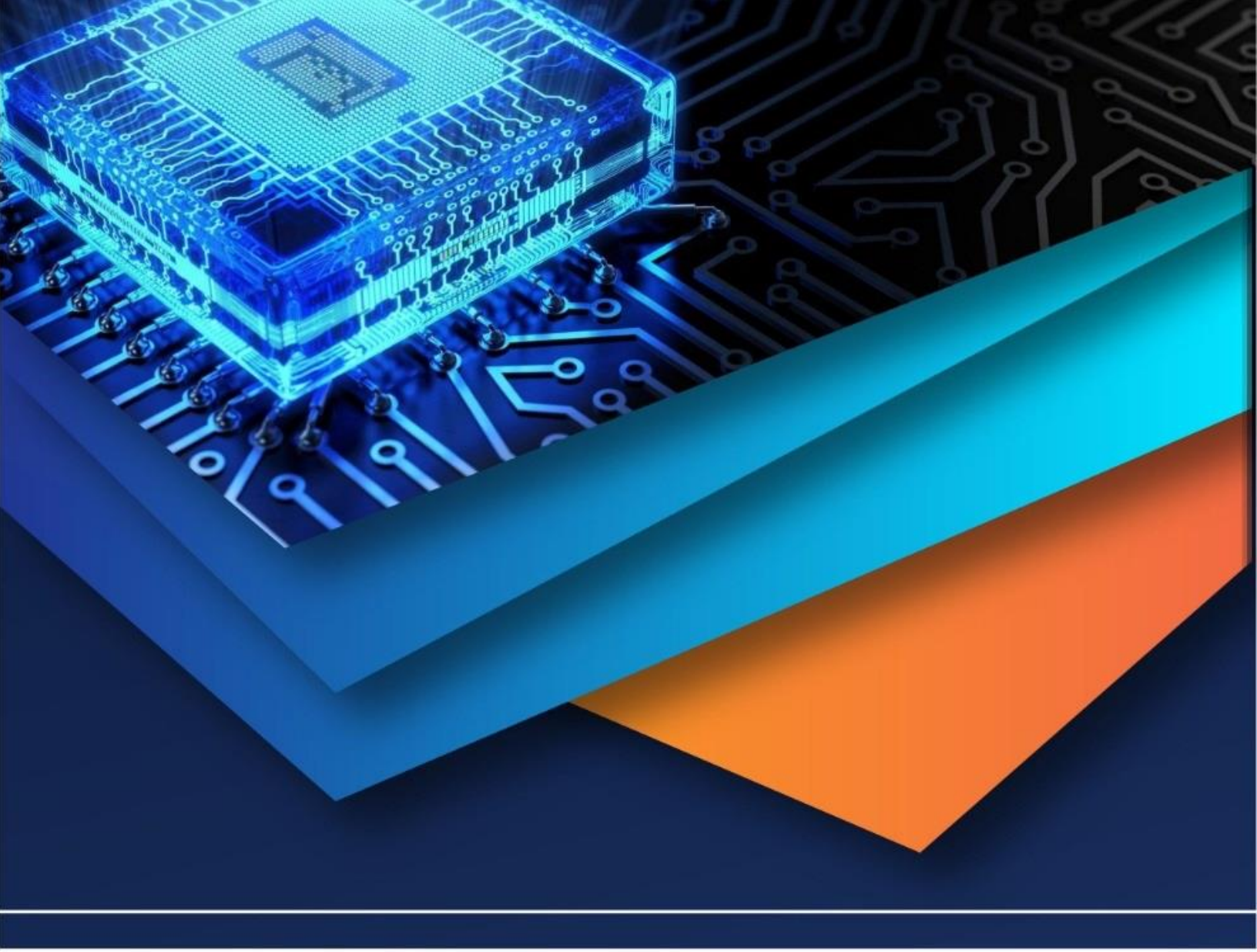
It can conduct subdomain enumeration operations with an accuracy of 95%, port scanning at 100% accuracy and shorten execution time by up to 40% as compared to traditional flows. Other features that the tool possesses such as the ease of performing large-scale scans while allowing for limited output define the importance of the tool to the cybersecurity sector.

Nonetheless, the external utilities also come with the constraints such as nonavailability of AI assisted prioritizing of the vulnerabilities, dependencies on online CVE databases for realistic mappings. These limitations act as encouragement for improving the existing systems and arising opportunities such are reinforcing the program with ML technology, advance layering technologies, offline mapping of vulnerability, implementing graphical user interfaces as well as advanced features to target wider audience.

To conclude, not only is Recon Automator an effective tool to use but it is also devoid of any weaknesses – its interface is user-friendly, it is accurate and efficient. With the help of automatic reconnaissance, security experts won't be spending unnecessary time addressing issues instead concentrating on finding solutions to the existing problems. Once fully developed, the program is likely to become one of the leading programs which undertake ethical hacking therefore strengthening security.

REFERENCES

[1] J. Scarfone, K. Ballintine, and M. Souppaya, Technical Guide to Information Security Testing and Assessment, National Institute of Standards and Technology (NIST), 2008.
[2] OWASP Foundation, "Amass," accessed November 2024, Available at:https://owasp.org/www-project-amass/.
[3] Nmap Organization, "Nmap," accessed November 2024, Available at: https://nmap.org/ .
[4] CVE Program, "Common Vulnerabilities and Exposures," accessed November 2024, https://cve.mitre.org/.
[5] Tib3rius, "Auto Recon: Automated Network Recon Tool," Auto Recon GitHub Repository, n.d., http://https://github.com/Tib3rius/AutoRecon,
[6] K. Kaur and S. Sharma, M. Singh, "A Study on Ethical Hacking Tools for Reconnaissance," International Journal of Advanced Research in Computer Science, vol. 9 no. 2, 2018, pp. 100-104.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY