Sudomy: Information Gathering Tools for Subdomain Enumeration and Analysis

Rizdqi Akbar Ramadhan¹, Redho Maland Aresta², Dedy Hariyadi³

¹Islamic University of Riau, Indonesia.
²University of Amikom Yogyakarta, Indonesia.
³Universitas Jenderal Achmad Yani Yogyakarta, Indonesia.

¹rizdqiramadhan@eng.uir.ac.id, ²screetsec@gmail.com, ³dedy@unjaya.ac.id

Abstract. In general, system security has become a crucial element in the digital aspect. Technically, evaluating the system there are aspects, one of which is by conducting a security assessment. Specifically, each system that will be evaluated is the essence of vulnerability search. The information security assessment is a form of awareness regarding cyberattacks that always increases from year to year. The assessment process can be carried out by an internal team and/or external auditor. The stages of evaluation by internal teams are certainly different from those of external parties. External auditors in conducting assessments need to learn or get as much information as possible related to the target, which in this case is in the form of a subdomain. Therefore, we need applications that support effective and efficient Information Gathering to assist in analysis and reporting. There are still many Information Gathering applications that do not include reporting and data validation systems. In this study, it is proposed to develop applications to support the Information Gathering stage which makes it easier for Cyber Security researchers/analysts.

1. Introduction

Based on the Indonesia Security Incident Response Team on Infrastructure / Coordination Center (ID- SIRTII / CC) that the level of internet security in Indonesia in 2018 has a poor ID-SIRTII Index value. When compared with Japan through the Mejiro Analysis has a higher risk, see Figure 1 even the total cyberattacks during 2018 totalling 232,447,974 attacks. The attack was monitored by ID-SIRTI / CC from January to December 2018 [1]. Bad predicate on internet security in Indonesia in 2018 needs to be introspected by network infrastructure managers that are connected to the internet. Therefore it is necessary to conduct security testing of information technology infrastructure on a regular basis to determine vulnerabilities in systems and information technology that can harm an organization [2]. To avoid cyber attacks on information technology infrastructure, the Ministry of Communication and Information has issued Ministerial Regulation No. 4 of 2016 which regulates the information security management system by electronic system administrators. The regulation stipulates that every electronic system operator must implement SNI ISO / IEC 27001 and/or Information Security Index guidelines [3].
Testing of information technology infrastructure security systems can be carried out by experts in the field of information security (pentesters). The types of tests carried out by the pentester are 6 types [2]:

- **Blind**, the target party has prepared the system to be tested while the pentester when testing does not know the detailed information related to the target system to be tested.
- **Double-Blind**, the target party does not prepare the system to be tested while the pentester when testing does not know the detailed information related to the target system to be tested.
- **Gray Box**, the target party has prepared a system that will be tested while the pentester will conduct testing when given simple information by the target party related to the system to be tested.
- **Double Gray Box**, the target party has prepared the system to be tested while the pentester will conduct testing when given the scope of testing by the target party related to the system to be tested.
- **Tandem**, the target party and the pentester prepare together regarding the target system to be tested.
- **Reversal**, the pentester is fully aware of the operational and target work systems. The implementation of the test is unknown to the target party. Both the target and the pentester are from the same organization.

Penters with Blind and Double-Blind types did not realize the details of the system to be tested. Unlike the Gray Box type, Double Gray Box, Tandem, and Reversal already know the details of the system to be tested. Therefore the Blind and Double-Blind pentester types require tools to find out the basic or initial information of a system to be tested. To get the basic information needed by the pentester, in this study we built Sudomy, an information-gathering tool. Sudomy was built to complement the tools needed by the pentester following the rules of the National Institute of Standards and Technology (NIST) and/or the Information Systems Security Assessment Framework (ISSAF).
2. Related Works

2.1. Domain Name System

Domain Name System (DNS) is a system that functions to convert a domain name that is easy to remember in the form of an IP Address by requesting information to a system that has a hierarchy and spread. The existence of DNS makes it easier to connect computing resources both through the internet and internal networks [3]. The global implementation of the DNS system is the implementation of three roles, namely the Domain Name Registry Operator, Domain Name Registrar, and Service Providers and Customers. The overview of the Domain Name Registry can be seen in Figure 2 [4].

![Figure 2. Overview of Domain Name Registry](image)

2.2. National Institute of Standards and Technology SP 800-42

The US Department of Commerce published recommendations on Network Security Testing as set out at the National Institute of Standards and Technology Special Publication 800-42 (NIST SP 800-42). The basic methodology for penetration testing according to NIST SP 800-42 consists of four phases, namely Planning, Discovery, Attack, and Reporting, see Figure 3 [5]. In the initial phase of Discovery, the pentester can identify and gather the information that is potentially related to the target. Information gathering can be carried out various techniques including Domain Name System interrogation, InterNIC queries, Search of the target organization's web server(s) for information, Search of the organization's Lightweight Directory Access Protocol server(s) (LDAP) for information, Packet capture, NetBIOS enumeration, Network Information System, and Banner grabbing.

![Figure 3. NIST: Penetration Testing Methodology](image)
2.3. Information Systems Security Assessment Framework (ISSAF)

The Information Systems Security Assessment Framework (ISSAF) has three methodological phases, namely Planning and Preparation, Assessment, and Reporting, Clean-up, Destroy Artefacts. Whereas in the Assessment phase there are nine steps: Information Gathering, Network Mapping, Vulnerability Identification, Penetration, Gaining Access & Privilege Escalation, Enumerating Further, Compromise Remote Users / Sites, Maintaining Access, and Covering Tracks, see Figure 4 [6].

![ISSAF Penetration Testing Methodology](image)

**Figure 4. ISSAF: Penetration Testing Methodology**

In the Information Gathering step, finding information from the target can be done technically, non-technically, or a combination of both. This aims to get information that is potentially related to the gap of the target. Information Gathering can be divided into two parts, namely passive and active. In the Passive Information Gathering section, the techniques used to obtain information are not directly related to the target. It could also be that the Passive Information Gathering section uses a third party to obtain information. Whereas the Active Information Gathering section in the process of getting information is directly related to the target. So this in some countries declared illegal [6].

2.4. Web Application Programming Interfaces

The advantage of developing using Web Application Programming Interfaces (Web APIs) is that it speeds up the development of an application. Therefore the implementation of Web APIs in the span of 2005-2013 is always increasing, see ‘figure’ 5 [7]. Web APIs use the HTTP / HTTP protocol in communicating with major applications with Web APIs providers [8]. Passive Information Gathering application development utilizes Web APIs to speed up getting information from third parties, such as SecurityTrails, BinaryEdge, VirusTotal, Censys, and Shodan.
2.5 Bash Script

Bash (Bourne-Again Shell) Script [9] can be chosen as a simple and easy-to-understand interface for users while choosing the Python programming language because it supports object-oriented programming [10] and has library support that facilitates development. The use of Bash Script has three main components: lexical analysis and parsing, text expansion, and command execution see 'figure 6' [11]. Input can start from the interactive interface through the console or Bash Script file. The first component is the lexical analysis and parses each command into the data structure. Then the second component performs a series of expansions and replaces the variables of each command parsing by following the sophisticated rules. Each command execution is then interpreted.

3. Tools & Methods

3.1 System Design and Testing of Sudomy

Development of Information Gathering applications follows ISSAF rules [6] by applying two techniques, namely passive and active. Passive techniques obtain information through a number of ways by utilizing third-party resources such as using Web APIs, Information Gathering libraries or through OSINT Source [12] with scraping processes. While the active technique uses applications that are installed with similar features, namely the Information Gathering function either by brute force, word lists or other new methods. The sudomy application diagram in conducting the Information Gathering stage can be seen in 'figure 7'. The Passive Information Gathering section uses resources from third parties using either the Web API, library or web scraping. The web scraping time testing process uses the regular expression (regex) technique on HTTP Body which uses three tests, namely the utilization of Native Apps that are called through Bash Script and libraries in two programming languages (Python and Go). The fastest time when the regex process uses Native Apps via Bash Script, which is Curl. The regex test results on the HTTP Body OSINT Source can be seen in Table 1.
Figure 7. Overview of Sudomy Application

Table 1. RegEx HTTP Body Comparation

| Command       | Native/Library                  | Real  | Users   | System (CPU) |
|---------------|---------------------------------|-------|---------|---------------|
| time ./curl.sh| curl (Native)                   | 0m0.189s | 0m0.032s | 0m0.017s     |
| time ./curl.py| import requests (Python)        | 0m0.693s | 0m0.318s | 0m0.110s     |
| time ./curl.go| package req (Go)               | 0m0.308s | 0m0.016s | 0m0.010s     |

Whereas the Active Information Gathering section utilizes the Native Apps Gobuster. Gobuster application does not have a graphical interface, so it is easier to combine on sudomy which both have Command-Line Interface (CLI) interfaces. The technique used in Active Information Gathering is the brute force on a target’s directory, DNS, and Virtual Host.

3.2 Testing Materials

In testing Information Gathering tools using several supporting devices both hardware and software. The device used is a notebook with Intel® Core™ i7-7700HQ @ 2.80Ghz CPU specifications and 8GB DDR4 RAM, with Virtual Box virtualization software running on the MS Windows 10 operating system. Virtual Box has a 2 core CPU configuration and 4GB RAM installed Kali Linux operating system. The Information Gathering process requires an internet connection, the provider used is PT Telekomunikasi Indonesia. The material details needed in the research are shown in Table 2.

Table 2. Materials

| Materials     | Description                                                                 |
|---------------|-----------------------------------------------------------------------------|
| Notebook      | ASUS ROG GL553VD (CPU: Intel® Core™ i7-7700HQ @ 2.80Ghz; RAM: 8GB DDR 4)     |
| Software      | Virtual Box                                                                 |
| Host OS       | MS Windows 10                                                               |
| Guest OS      | Kali Linux (RAM 4GB; No DE; Using Windows Manager)                          |
| Ping Connection| 2 ms                                                                        |
| Download/ Upload | 30 Mbps / 7 Mbps / 30 Mpbs / 7 Mbps                                         |
| Provider      | PT Telekomunikasi Indonesia Tbk (Telkom)                                    |
4. Result and Discussion

Information Gathering Method in sudomy application, there are two Passive and Active techniques. The passive technique uses third party resources such as DNSdumpster, WebArchive, Shodan, Total Virus, Certsh, BinaryEdge, SecurityTrails, Certspotter, Censys, Threatminer, Bufferover, Hackertarget, Entrust, ThereatCrowd, and Riddler. To improve the enumeration results sudomy application needs to add an API Key for Shodan, Censys, Total Virus, BinaryEdge, and SecurityTrails in the sudomy.API section. Whereas Active technique uses a combination of the Gobuster application with the wordlist provided by SecLists. SecLists has a collection of approximately three million wordlists. Testing is done by comparing the sudomy application with other applications such as sub finder and sublist3r with the target domain bugcrowd.com. The sub finder application uses 25 resources, sublist3r uses 11 resources, and sudomy uses 16 resources. The time needed to search for a subdomain from bugcrowd.com, the sub finder application takes 1 minute 38,621s, sublist3r takes 0 minutes 27,216s, and sudomy takes 0 minutes 6,946s. The subdomain results from bugcrowd.com found by the sub finder application are 25 subdomains, sublist3r are 23 subdomains, and sudomy is 49 subdomains. The results of enumeration looking for the bugcrowd.com subdomain using the sub finder, sublist3r, and sudomy applications can be seen in table 3. To further speed up the enumeration process and save CPU, RAM, and bandwidth, you can use third party resources as needed.

| Tools   | Time            | Resources (Source) | Results (Subdomain) |
|---------|-----------------|--------------------|---------------------|
| subfinder | 1m 38.621s     | 25                 | 25                  |
| sublist3r | 0m 27.216s    | 11                 | 23                  |
| sudomy   | 0m 6.946s       | 16                 | 49                  |

The sudomy application has been equipped with a reporting system with HTML and CSV output format that makes it easy for Cyber Security researchers and / or analysts. The HTML formatted report displays subdomain enumeration results including Ping Sweep, HTTP status, IP Address, top ten open protocols, TakeOver Subdomain checking, screenshots, httprobe (utility for HTTP / HTTPS validation) and graphs resulting from enumeration, see Figure 8.

![Figure 8. Dashboard Report Sudomy](image-url)
5. Conclusion

Based on testing with similar tools, sudomy is still faster in the enumeration process of searching subdomains. Besides that, Sudomy also has features that really help Cyber Security researchers and/or analysts. The features that can be used include: Host checking on active subdomains, HTTP status code response information, conversion from subdomain list to IP Resolver, port scanning from IP Resolver, checking of TakeOver Subdomain attacks, screenshots through domain list, DNS Bruteforce Subdomain attack, and reporting. In addition, sudomy application can be downloaded freely at https://github.com/Screetsec. Further development is expected to be able to integrate sudomy with other information security assessment applications.

Acknowledgement

Authors would like to say thank you very much to KEMENRISTEKDIKTI Indonesia and Universitas Islam Riau, University of Amikom Yogyakarta, Universitas Jenderal Achmad Yani Yogyakarta, Indonesia for funding this research project and support the facilities.

References

[1] Indonesia Security Incident Response Team on Infrastructure / Coordination Center 2019, Indonesia Cyber Security Monitoring Report 2018 (Jakarta), 2018.

[2] Institute for Security and Open Methodologies, The Open Source Security Testing Methodology Manual, 2010.

[3] P. V. Mockapetris, RFC 1035: Domain Names - Implementation and Specification, 1987.

[4] A. A. Sinaci, O.T. Sehitoglu, M. T. Yondem, G. Fidan and I. Tatli, SEMbySEM in Action: Domain Name Registry Service Through a Semantic Middleware eChallenges, 2010 1–8, 2010.

[5] K. Scarfone, M. Souppaya, A. Cody and A. Orebaugh, NIST SP 800-42: Guideline on Network Security Testing vol 115 (Gaithersburg, MD), 2015.

[6] Open Information Systems Security Group, Information Systems Security Assessment Framework (ISSAF), 2006.

[7] C. Rudrakshi, A. Varshney, B. Yadla, R. Kanneganti and K. Somalwar, API-fication, 2014.

[8] A. Mendoza, and G. Gu, Mobile Application Web API Reconnaissance: Web-to-Mobile Inconsistencies & Vulnerabilities Proc. - IEEE Symp. Secur. Priv. 2018-May 756–69, 2018.

[9] C. Ramey and B. Fox, Bash Reference Manual (Free Software Foundation, Inc.), 2019.

[10] D. Hariyadi and Fazlurrahman, Membangun Telegram untuk Crawling Malware OSINT Menggunakan Raspberry Pi Indones. J. Bus. Intell. 2 18–24, 2019.

[11] I. J. Davis, M. Wexler, C. Zhang, R.C. Holt and T. Weber, Bash2py: A bash to Python translator 2015 IEEE 22nd Int. Conf. Softw. Anal. Evol. Reengineering, SANER 2015 - Proc. 508–11, 2015.

[12] F.E. Nastiti, D. Hariyadi and Fazlurrahman, TelegramBot : Crawling Data Serangan Malware dengan Telegram J. Comput. Eng. Syst. Sci. 4, 2019.