# 1. INTRODUCTION

## 1.1 MOTIVATION

The explosive growth of the web has brought many goodies like E-commerce, E-mail, Cloud computing, but there's also a black side of Hacking, malwares etc. Hacking is one of the biggest problems faced by tech companies, governments, and citizens across the world. An Ethical Hacker or a Penetration tester can assist or help the people that are suffering from these cyber-attacks. Ethical hacking is usually termed as online geeks or groups that legally access the company's online assets after obtaining official approval. Reconnaissance means the preparatory stage where an Ethical Hacker seeks to collect the maximum amount of information as possible a few targets before launching a security test.

It involves 3 phases namely, scanning, foot printing, and enumeration of the organization's network. In this project, we'll be handling automation of foot printing of an organization. Foot printing is nothing but the blueprint of the safety of a corporation, undergone during a procedural manner. It finds all information available on the internet about the target. It is a time-taking process to flick through the sites and collect info; hence in this paper, we investigate the solution for tedious web search and propose a proficient way to organize, extract and store data from the search engines employing a new tool, Search Simplified. Info gathering techniques are often broadly divided into the following:

Active: This includes intrusive recon that sends (specially built) data to the target, for example, port scanning. Advanced network foot printing techniques dodge direct connections with the target host.

Passive: This is the kind of reconnaissance that either does not contact or communicate directly to the target system or that uses publicly available information, and not normally found from standard logs. This paper also focuses on this technique.

Both active and passive reconnaissance can cause the invention of useful data to use in a malicious activity. This information may enable an attacker to seek out vulnerabilities in the OS's version and exploit the loophole to gain more access. Shell-script based Recon Framework is a fully featured recon framework which is written in shell script.

Ethical hacking begins with gathering information and becoming familiar with the target system. Reconnaissance refers to a set of processes and techniques, such as footprinting and scanning and enumeration that are used to gather and covertly discover as much information as possible about a target system.

Reconnaissance is an essential step in locating and stealing confidential information. In a proper recon, attackers would have access to detailed information. In this way, reconnaissance, in information security, is used for penetration testing.

To gain information without actively engaging with the network, an attacker uses recon to interact with the network's open ports, running services, etc. The information it provides can help gain access to networks beyond the internet. In short, recon is a treasure trove of valuable information that is susceptible to attacks.

Unexpectedly, it is unknown how long a recon can take to get into networks; it may take weeks or months. Moreover, a recon may not access any information system but still result in data breach, collecting all sensitive data at once, exploiting networks.

An ethical hacker takes the following seven steps during reconnaissance to gather as much information about a target system as possible:

- Collecting initial information
- Determining the network's range
- Identifying active machines
- Discovering available access points and ports
- Identifying the operating system by its fingerprint
- Locating services on ports
- Creating a network map

In order to gain information about a network, an attacker will use the following steps:

- File permissions
- Running network services
- OS platform

- Trust relationships
- User account information

Types of Reconnaissance

There are two main types of reconnaissance, active reconnaissance, and passive reconnaissance. Let us understand the difference between active reconnaissance and passive reconnaissance.

**Active Reconnaissance**

Cybercriminals, who use active reconnaissance, try to obtain information about computer systems using tools such as automated scanning and manual testing, ping, and netcat. Since active reconnaissance creates more noise within the system and has a higher chance of detection, it is generally faster and more accurate.

**Port Scanning**

Port scanning is an example of active reconnaissance. Port scanning is the process of scanning computer ports to identify open ports to a computer since the entire information is going in and out through these ports. Using port scanning, attackers determine what services are visible and where an attack can be conducted. As part of port scanning, data is retrieved from opened ports and analyzed.

**Usage in Penetration Testing**

Cyber reconnaissance is an integral part of penetration testing. It is this step that dictates what is going to be done in the subsequent steps of the test. As part of the reconnaissance, we can use passive information gathering techniques to gather information about a company, its employees, and the technology it uses. Information can also be gathered by using active information gathering techniques about specific systems of the target, such as operating system, services run on the machine, and open ports. An effective penetration tester will utilize both types of information gathering to find the best method of breaching a company.

## 1.2 PROBLEM DEFINITION

Ethical hacking provides a way of finding vulnerabilities in web sites and object-oriented programs that are connected with internet or cyber stations. The main problem is how to protect the data in cyber world that can be theft by the hackers. Ethical hacking and penetration testing is slowly increasing its demand in India, as a result of heists, the cyber-hacks are rapidly increasing in India due to vulnerabilities present in the websites. Indian websites are extremely prone to Cross - Site Script attacks, which can further lead to web defacement. While making a statement about Digital India, the honorable Prime Minister, Mr. Narendra Modi Criticized that the Indian Websites are easily prone to cyber-attacks from the enemy countries due to lack of cyber security experts in India. Ethical hacking is the field which is growing exponentially in India.

Every IT organization, including the tech giants, which is dealing with users and their information with privacy, needs an ethical hacker to protect their network and domain. Ethical hacking is not just the use to automated tools against web apps or servers, it's deeply about how you can pre-test a project manually using your out-box thinking, tools are just a way to perform your action.

The increase in number and intricacy of cyber-attacks is the cause due to which these attacks remain undetected, and their number stretches to thousands per day. In order to thwart such attacks the virtual security programmers are putting all their endeavors to protect databases, computer programs, systems and networks from such attacks that include unauthorized access, spoofing or obliteration.

Moreover, it is also a serious problem for government and policy makers to control this abject condition of security across the world. In case of corporate and e-service sector, capital markets and other businesses functions are based on large amount of confidential data stored on computers and are shared on internet for their specialized purposes.

Reconnaissance attacks are used to gather information about a target network or system. Such attacks may seem harmless at the time and may be overlooked by security administrators as "network noise" or pestering behavior, but it is usually the information gained through reconnaissance attacks that is used in subsequent Access or DoS attacks.

Several means may be used to gather information about an organization and could include automated and manual technological attacks as well as human social attacks. Examples might include ICMP ping sweeps against a network or SNMP walking techniques to gather network map and device configuration data

Active reconnaissance includes interacting directly with the target. It is important to note that during this process, the target may record IP address and log activity. Passive reconnaissance makes use of the vast amount of information available on the web. When one is conducting passive reconnaissance, one is not interacting directly with the target and as such, the target has no way of knowing, recording, or logging activity.

The reconnaissance is aimed at collecting as much information as possible on a target. At this point in the penetration test, no detail should be overlooked regardless of how innocuous it may seem. While one is gathering information, it is important to keep the data in a central location. Reconnaissance begins by closely reviewing the target's website. In some cases, a tool called HTTrack is used to make a page-by-page copy of the website. The copied website will include all the pages, links, pictures, and code from the original website; however, it would reside on local computer. An excellent tool to use in reconnaissance is The Harvester.

## 1.3 OBJECTIVE

The objective of this project to collect as much information as possible. Collecting information and knowing deeply about the target system is known as “Reconnaissance”. This data is the main street for the programmer to hack the target system. It involves Foot printing, Enumeration, and Scanning.

In the tools that we are likely to see used in passive reconnaissance, we will find various scanning tools, such as network sniffers for both wired and wireless networks, port scanners, vulnerability analysis tools, operating system fingerprinting tools, banner grabbing tools, and other similar utilities. We will be looking to enumerate the infrastructure devices, networks, and systems in place in the environment; assess the ports open and services operating on those ports; fingerprint operating systems; and assess vulnerabilities. This process is certainly not set in stone and is intended as a general guideline. There will be times when a chain of interesting information will lead us to one

step sooner than another and there is absolutely nothing wrong with varying the approach.

- Service Enumeration, where you identify the services running on a server, and determine any vulnerability they might have.
- Assess Vulnerabilities, where you identify vulnerabilities in an app, site, or network. You might use a vulnerability database, knowledge bases, and a vulnerability scanner like Open VAS to scan a system and provide a report.
- Exploit Vulnerability, where you either find an existing exploit or develop a new one that can take advantage of vulnerabilities you've discovered.
- Conducting reconnaissance using publicly available data sets.
- Automating bug bounty hunting using OSINT.
- How to conduct reconnaissance efficiently and effectively.
- Check/Verify target's scope (*.example.com).
- Find subdomains of target (Refer Subdomain tools mentioned in the article).
- Check which domains resolve.
- Take Screenshot.

Active reconnaissance is a type of computer attack in which an intruder engages with the targeted system to gather information about vulnerabilities.

## 1.4 LIMITATIONS

The word reconnaissance is borrowed from its military use, where it refers to a mission into enemy territory to obtain information. In a computer security context, reconnaissance is usually a preliminary step toward a further attack seeking to exploit the target system. The attacker often uses port scanning, for example, to discover any vulnerable ports. After a port scan, an attacker usually exploits known vulnerabilities of services associated with open ports that were detected.

Somewhat confusingly, active, and passive reconnaissance are both sometimes referred to as passive attacks because they are just seeking information rather than actively exploiting the targets, as active attacks do.

Both active and passive reconnaissance are also used for ethical hacking, in which white hat hackers use attack methods to determine system vulnerabilities so that problems can be taken care of before the system falls prey to a real attack.

The simplest way to prevent most port scan attacks or reconnaissance attacks is to use a good firewall and intrusion prevention system (IPS). The firewall controls which ports are exposed and to whom they are visible. The IPS can detect port scans in progress and shut them down before the attacker can gain a full map of your network.

Several means may be used to gather information about an organization and could include automated and manual technological attacks as well as human social attacks. Examples might include ICMP ping sweeps against a network or SNMP walking techniques to gather network map and device configuration data

Active reconnaissance includes interacting directly with the target. It is important to note that during this process, the target may record IP address and log activity. Passive reconnaissance makes use of the vast amount of information available on the web. When one is conducting passive reconnaissance, one is not interacting directly with the target and as such, the target has no way of knowing, recording, or logging activity.

The reconnaissance is aimed at collecting as much information as possible on a target. At this point in the penetration test, no detail should be overlooked regardless of how innocuous it may seem. While one is gathering information, it is important to keep the data in a central location. Reconnaissance begins by closely reviewing the target's website. In some cases, a tool called HTTrack is used to make a page-by-page copy of the website. The copied website will include all the pages, links, pictures, and code from the original website; however, it would reside on local computer. An excellent tool to use in reconnaissance is The Harvester. the main street for the programmer to hack the target system. It involves Foot printing, Enumeration, and Scanning.

In the tools that we are likely to see used in passive reconnaissance, we will find various scanning tools, such as network sniffers for both wired and wireless networks, port scanners, vulnerability analysis tools, operating system fingerprinting tools, banner grabbing tools, and other similar utilities. We will be looking to enumerate the infrastructure devices, networks, and systems in place in the environment; assess the ports open and services operating on those ports; fingerprint operating systems; and assess vulnerabilities.

# 2. LITERATURE SURVEY

Reconnaissance, also known as information gathering, is classified as active and passive reconnaissance. Active reconnaissance includes interacting directly with the target. It is important to note that during this process, the target may record IP address and log activity. Passive reconnaissance makes use of the vast amount of information available on the web. When one is conducting passive reconnaissance, one is not interacting directly with the target and as such, the target has no way of knowing, recording, or logging activity. The reconnaissance is aimed at collecting as much information as possible on a target. At this point in the penetration test, no detail should be overlooked regardless of how innocuous it may seem. While one is gathering information, it is important to keep the data in a central location.

Reconnaissance begins by closely reviewing the target's website. In some cases, a tool called HTTrack is used to make a page-by-page copy of the website. The copied website will include all the pages, links, pictures, and code from the original website; however, it would reside on local computer. An excellent tool to use in reconnaissance is The Harvester. The Harvester is a simple but highly effective Python script written by Christian Martorella at Edge Security. This tool allows quickly and accurately catalogs both e-mail addresses and subdomains that are directly related to the target.

Reconnaissance is the first step a hacker will take, where they try to gather as much information as possible about a target. Often, a hacker will begin with passive reconnaissance, which doesn’t involve direct interaction, is harder to detect, and doesn’t involve using tools that touch the target’s site, network, or computers. Some of the ways you might do passive reconnaissance include Search engines, which may reveal documents with the names of a Virtual Private Network (VPN) the company uses, vendor documentation mentioning that the target is a client using certain products (routers, software, etc.). In doing this, you may get information on the company’s remote access, and see cache pages that allow you to stay passive. Job advertisements, which can reveal contact information, requirements to know certain software or equipment that may have vulnerabilities that can be exploited, and so on. LinkedIn and other

sites where employees have identified their involvement with a target. Who is sites (like www.who.is) that provide the names of servers, IP address ranges, the names of administrators, email addresses, and so on. Way back Machine (www.archive.org) to see past versions of a website, allowing you to review the target's site, see contact information for employees, and even content that may have deemed a security risk and removed.

Once you've learned what you can do without touching a site or network, a hacker will move onto active reconnaissance, which involve interaction with a target and could be traceable. For example, a hacker may call or talk to employees, visit their website, or other actions in which they touch the network as a normal user. After gathering everything you can on a company, its infrastructure, personnel, and other details that can help you gain access, you should have a good idea of the company's structure and network, and ready to move onto other steps, scanning, where you try and identify what hosts are live and their purpose on a network. The hacker might use the PING command to see what servers are running or use port scanning software to find weaknesses like open ports or ways to bypass firewalls. In doing so, he or she may throttle the scan, so it's slow pings and scans hide in the normal network traffic and isn't easily detectable. Service Enumeration, where you identify the services running on a server, and determine any vulnerability they might have. Assess Vulnerabilities, where you identify vulnerabilities in an app, site, or network. You might use a vulnerability database, knowledge bases, and a vulnerability scanner like Open VAS (www.openvas.org) to scan a system and provide a report. Exploit Vulnerability, where you either find an existing exploit or develop a new one that can take advantage of vulnerabilities you've discovered.

At this point, the hacker is finally at a stage where he or she can use the gathered information to attempt breaking into a system or site. The method used will depend on the skill level of the hacker, and what's easiest and makes the most sense to achieve their goals. For example, if they can get a username and password for an FTP site from a list or through social engineering, they might logon as an authentic user, and then modify or upload web pages so the content is different. If they've accessed an administrator account, they have full control of the server or system. If not, they may try to exploit vulnerabilities they've

found to elevate their privileges to this level.

A hacker's goal may be to roam the folders and file structure of a server, hoping to find documents and data sources that make their expedition fruitful, but they may not even need to go this far. If a Web developer has poorly coded an application, an administrator hasn't set permissions properly, and/or security is ineffective, a hacker might be able to do what he or she wants using forms that accept user input or URLs that accept parameters.

Depending what the hacker hoped to achieve, he or she may leave a site, so it appears untouched. Any evidence of what was done is cleaned up, permissions are reset, and log files are deleted. The hacker may install a toolkit (which we discuss later) or other malware; he or she may create fake accounts or backdoors that will allow him or her future access. By the end of it, administrators and users may be fooled into believing nothing's happened, and they're safe.

Despite the appearance, a hacker might modify links to spread malware or redirect users to a site they control. Click jacking is a method of adding a hyperlink to clickable content. You might visit a page, and a popup window appears that you try and close by clicking an "X" button. Because the link's been modified, clicking it may actually download a Trojan Horse, transfer money from an account on the site you're visiting, send you to another site to gather information from you, or some other action you didn't expect.

A variation on click jacking is like jacking, where you'll see a post or status update on Facebook. It is a common venue for click jacking, where it often takes the form of like jacking. A post or status update may promise a video or have an intriguing or scandalous draw. When you click on it, you might be asked to like or share the post before you 've even seen it, presented with a fake CAPTCHA or a link that asks you to take a test to prove you're human. However, these aren't actually challenges to prove you're not a robot. Links and buttons on the page will run code to share or like the posts, distributing the spam to others viewing your posts. These scam posts are often used to gather user information, and may redirect you to other spam, phishing, or other malicious websites.

While hacking a site may seem covert, many hackers will post information about it on the Internet. Details of the hack, samples of data, or links to a complete dump of the database may appear on sites like Paste in (www.pastebin.com), allowing others to view and download the data. Other sources of finding hacked data include Internet Relay chat, tweets about new dumps on Paste bin through Dump Monitor (@dumpmon), or Twitter accounts belonging to a person or group responsible for the data breach. This shares the information with other cybercriminals, who may then use the data to commit other crimes.

Doxing is another practice of sharing information acquired through hacking and other means. Dox is a homonym for docs (i.e., documents) and involves uploading sensitive documents or a dossier of information onto the Internet. For example, in March 2013, the personal and financial information of numerous celebrities were posted on a site called Expose.su. Some of the victims included FBI Director Robert Mueller, Kim Kardashian, Hillary Clinton, Mel Gibson, Ashton Kutcher, and others. Web pages on the site displayed such information as their full names, birthdates, Social Security numbers, current and previous addresses, phone numbers, and copies of a credit report. It was found that while some details on the site were false, other information was accurate groups.

While hackers may work alone, there are informal and organized communities of hackers who work together and share information. Those involved include elite hackers with programming and database skills, script kiddies who use other people's scripts and instructions because they lack the expertise to do it themselves, and those who fall in between who are developing their skills and evolving into a more sophisticated threat. For those involved, it may provide support, a sense of belonging, and a way for new hackers to find mentorship.

Modern technology has made it more difficult defending yourself against shoulder surfing. While you should be wary of anyone behind you or nearby, who may be watching what you're doing on a keyboard or screen, you probably won't notice someone watching you over a closed-circuit security camera or watching from a distance with binoculars.

Therefore, even though no one is in sight, don’t assume no one is watching.

Another useful tactic is dumpster diving, in which a person simply goes through your trash trying to get a hold of telling information. If it’s a business’ garbage, they may find printed maps of the network infrastructure, billing records, manuals, or employee names. This could be useful for social engineering, or (if someone threw out a sticky note with a password) hacking. If it’s your home garbage, they may find preapproved credit cards, a bill with account numbers, or other information to steal your identity. To avoid being a victim, you should shred any documents with sensitive or personal information, as well as ID, and financial cards.

If a hacker wanted access to the building so they could have direct access to employees, computers, and other devices, they might tailgate someone who works there. Tailgating involves following a person into a restricted area. The person may say that they forgot their keycard at their desk and fumble with a number of things to appear as if they’re having trouble finding their door card. If it was cold or raining, they might simply scurry inside behind you, complaining about the weather. Since people often want to be helpful, and don’t commonly challenge others if they appear confident and as if they belong there, tailgating is a simple and effective way of gaining entry.

An Internet reconnaissance test should be focused on assessing the organization's profile based on what information is publicly available on the Internet. Domain registries, the organization's financial statements, career postings, and vendor case studies are all sources of information about an organization that could be used by an attacker. Google has actually become a primary tool for would-be attackers to profile an organization looking for weaknesses that can be exploited by technical means or through social engineering. Any organization needs to have some level of public presence, a point that is emphasized by the introduction of the White House as an active participant on Facebook during the Obama administration.

The point of this type of testing is to have someone with the knowledge of typical data mining techniques look at the organization's profile from an Internet perspective and identify unnecessary information risks. Like other

passive testing methods, this assessment presents no risk of an operational disruption to the organization.

In recent years, the Internet has become an integral element of people's everyday lifestyles all across the world. Online criminality, on the other hand, has risen in tandem with the growth of Internet activity. Cyber security has advanced greatly in recent years in order to keep up with the rapid changes that occur in cyberspace. Cyber security refers to the methods that a country or organization can use to safeguard its products and information in cyberspace. Two decades ago, the term "cyber security" was barely recognized by the general public.

Cyber security isn't just a problem that affects individuals but it also applies to an organization or a government. Everything has recently been digitized, with cybernetics employing a variety of technologies such as cloud computing, smart phones, and Internet of Things techniques, among others. Cyber-attacks are raising concerns about privacy, security, and financial compensation. Cyber security is a set of technologies, processes, and practices aimed at preventing attacks, damage, and illegal access to networks, computers, programmes, and data. The primary goal of this article is to conduct a thorough examination of cyber security kinds, why cyber security is important, cyber security framework, cyber security tools, and cyber security difficulties.

Cyber security safeguards the data and integrity of computing assets that are part of or connected to an organization's network, with the goal of defending such assets from all threat actors throughout the life cycle of a cyber-attack.

A bug bounty is a monetary reward given to ethical hackers for successfully discovering and reporting a vulnerability or bug to the application's developer. Bug bounty programs allow companies to leverage the hacker community to improve their systems' security posture over time continuously.

Hackers around the world hunt bugs and, in some cases, earn full-time incomes. Bounty programs attract a wide range of hackers with varying skill sets and expertise giving businesses an advantage over tests that may use less experienced security teams to identify vulnerabilities. Bounty programs often complement regular penetration testing and provide a way for organizations to test their applications' security throughout their development life cycles.

Businesses starting bounty programs must first set the scope and budget

for their programs. A scope defines what systems a hacker can test and outlines how a test is conducted. For example, some organizations keep certain domains off-limits or include that testing causes no impact on day-to-day business operations.

This allows them to implement security testing without compromising overall organizational efficiencies, productivity, and ultimately, the bottom line. Bug bounties with competitive payouts tell the hacking community companies are serious about vulnerability disclosure and security. Programs base reward levels on the severity of vulnerabilities, and rewards increase as the potential impact increases.

## 2.1 EXISTING SYSTEM

The existing models available today have very limited features and are not compatible with the modern web development frameworks like MEAN and MERN stacks, Django, Flask or Spring Framework. As these new technologies and tech stacks came into limelight, there are many things which are overlooked and often much vulnerability are missed when tried with the existing recon frameworks. Some of the important things missed by the existing recon frameworks are:

- JavaScript file enumeration and analysis.
- Automation of Google Dorking.
- Automation of some known OWASP vulnerabilities like XSS, SSRF etc.
- Absence of project discovery's nuclei at the time of writing old recon frameworks.
- Automation of fuzzing for endpoints on the target.

Since the existing frameworks did not incorporate multithreading in their tools, the recon process takes a lot of time. The output management hasn't been up to the mark in any of the frameworks. And in addition to that, each recon framework lacks one or the other features like speed, accuracy, etc. These are the limitations of the existing models and thus there is a need for a fast and accurate framework which automates every single module of the information gathering phase.

## 2.2 DISADVANTAGES OF EXISTING SYSTEM

- Cases of Success and Remediation.
- Large Number of Unhelpful Alerts.
- Attracting Less or Wrong Talent.
- Less Focus on OS Vulnerabilities.
- Time Limit Issues.
- Public Reputation at Stake
- Low Possibility of Success and Income.

## 2.3 PROPOSED SYSTEM

Keeping in view the existing models, this proposed model is an attempt to overcome the limitations of the existing models and having updated tools and techniques which are mostly based on fingerprints of various endpoints of the target web application.

- The speed of the recon process is grater as the tools are written in Go and Python.
- Nuclei is another great tool in finding low hanging vulnerabilities.
- JavaScript enumeration is made simple than ever before.
- Dorking is now just one click away as all the main dorks for a target are incorporated in the project.
- With evolved wordlists, content enumeration is very effective with our project.
- Take the input (top-level domain) from the user as a command line argument to the recon script.
- Perform; subdomain enumeration on the target (top level domain name)
- Extract all the live subdomains which have a web server running on them from the enumerated subdomains list.
- Also gather the status codes and titles of the live subdomains.
- Perform google dorking on the subdomains.
- Get all the URLs once present on the target from way back machine.
- Perform credential stuffing on the target.

- Perform JavaScript enumeration on all the live subdomains.
- Perform fuzzing to find the hidden functionality and content on the subdomains.
- Perform a simple port scan to have an idea of what ports are open and what services are running on them.
- Perform nuclei scan on the target.
- Look for some simple vulnerabilities like Open Redirects, XSS, SSRF on some parameters obtained from the way back URLs.

## 2.4 ADVANTAGES OF PROPOSED SYSTEM

- Saves a lot of time.
- You know what exactly to look for.
- You can easily automate your recon workflow.
- Less of a chance to submit Out-of-Scope Issues.
- Just like other security methodologies, it enables you to perform better Recon.
- Increased Vulnerability Detection.
- Reduced Cost.
- Realistic Threat Simulation.

# 3. SYSTEM ANALYSIS

## 3.1 SOFTWARE REQUIREMENT SPECIFICATION

## 3.2 SOFTWARE REQUIREMENTS

- Operating System: LINUX, UNIX.
- Browser: Google Chrome, Mozilla Firefox.
- Frontend: Terminal, Bash Script, GO Language.
- Backend: Python(pip).

## 3.3 HARDWARE REQUIREMENTS

- RAM up to 4GB.
- Hard disk up to 100GB
- Pentium IV Processor.
- Input device: Keyboard, Mouse.
- Output device: Monitor.

# 4. SYSTEM DESIGN

## 4.1 SYSTEM ARCHITECTURE

The system architectural design is the design process for identifying the subsystems making up the system and framework for subsystem control and communication. The goal of the architectural design is to establish the overall structure of software system which is shown figure.

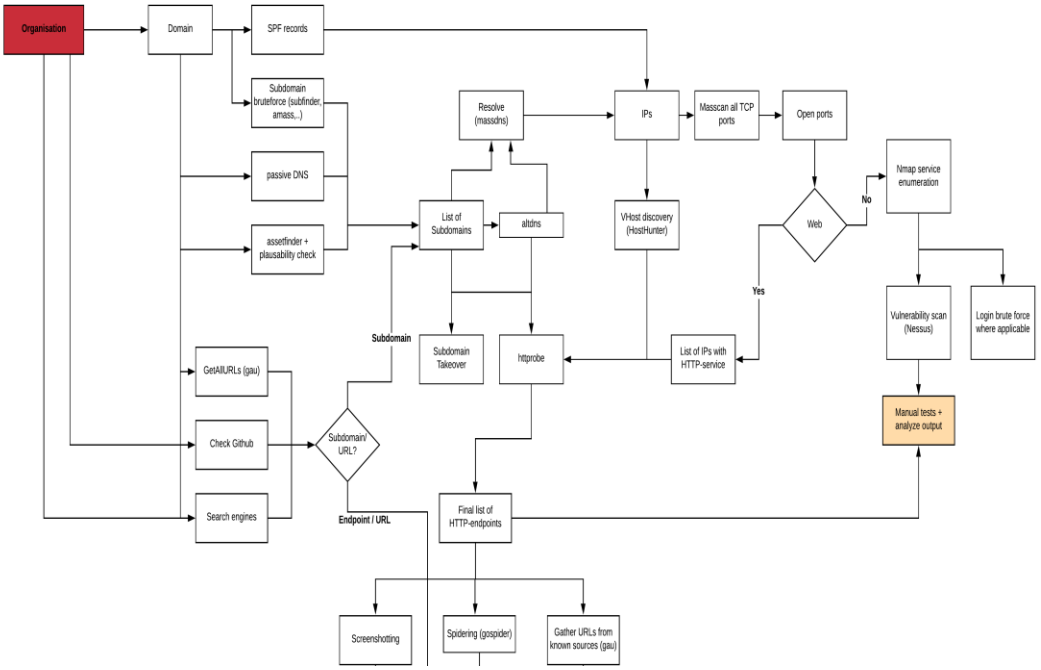


Figure 1: System Architecture for recon.

- The fundamental organization of a system, embodied in its components, their relationships to each other and to the environment, and the principles governing its design and evolution.
- A representation of a system, including a mapping of functionality onto hardware and software components, a mapping of the software architecture onto the hardware architecture, and human interaction with these components.

- An allocated arrangement of physical elements which provides the design solution for a consumer product or life-cycle process intended to satisfy the requirements of the functional architecture and the requirements baseline.
- An architecture consists of the most important, pervasive, top-level, strategic inventions, decisions, and their associated rationales about the overall structure (i.e., essential elements and their relationships) and associated characteristics and behavior.
- A description of the design and contents of a computer system. If documented, it may include information such as a detailed inventory of current hardware, software and networking capabilities; a description of long-range plans and priorities for future purchases, and a plan for upgrading and/or replacing dated equipment and software.
- A formal description of a system, or a detailed plan of the system at component level to guide its implementation.
- The composite of the design architectures for products and their life-cycle processes.
- The structure of components, their interrelationships, and the principles and guidelines governing their design and evolution over time.

System architecture conveys the informational content of the elements consisting of a system, the relationships among those elements, and the rules governing those relationships. The architectural components and set of relationships between these components that an architecture description may consist of hardware, software, documentation, facilities, manual procedures, or roles played by organizations or people.

System architecture primarily concentrates on the internal interfaces among the system's components or subsystems, and on the interface(s) between the system and its external environment, especially the user.

## 4.2 USE CASE DIAGRAM

Use case diagram represent the overall scenario of the system. A scenario is nothing but a sequence of steps describing an interaction between a user and a system. Thus, use case is a set Of scenarios tied together by some goal. This case diagram is drawn for exposing the functionalities of the system.

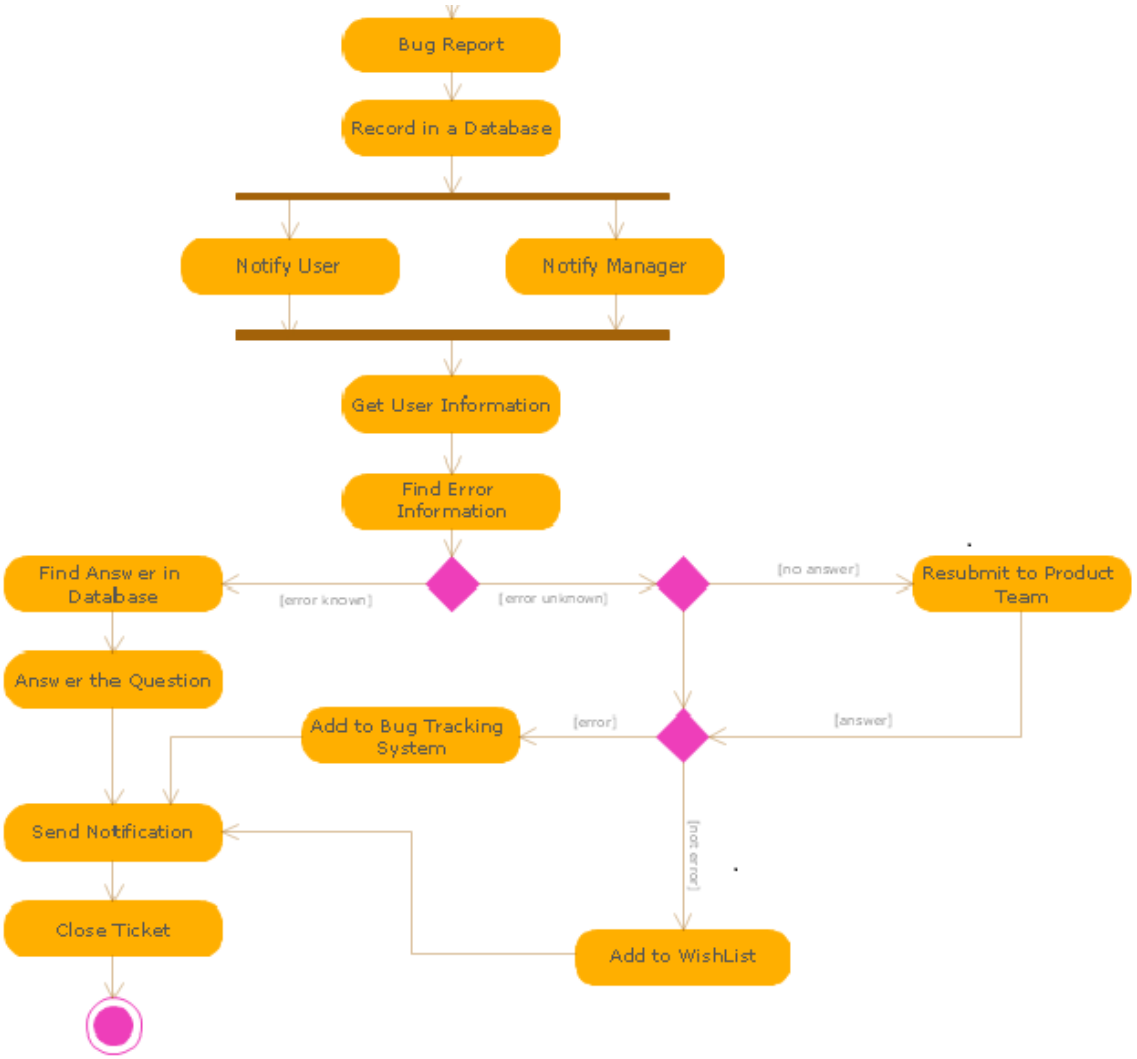


Figure 2: Use-case Diagram

## 4.3 ACTIVITY DIAGRAM

The activity diagram is a graphical representation for representing the flow of interaction within specific scenarios. It is similar to a flowchart in which various activities that can be performed in the system are represented.

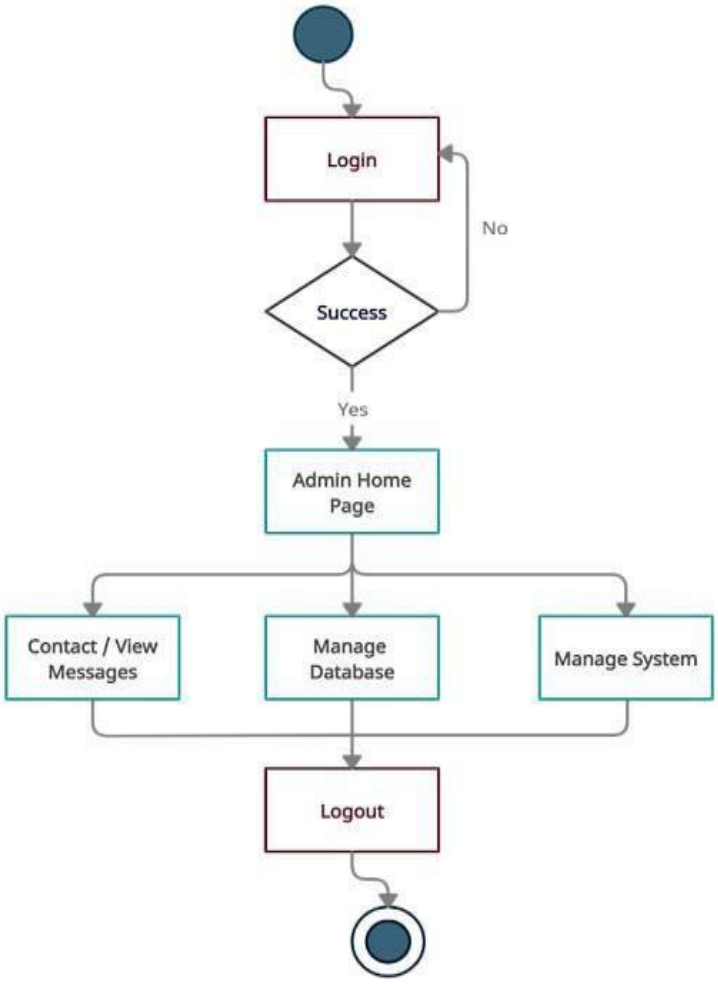


Figure 3: Root Activity Diagram

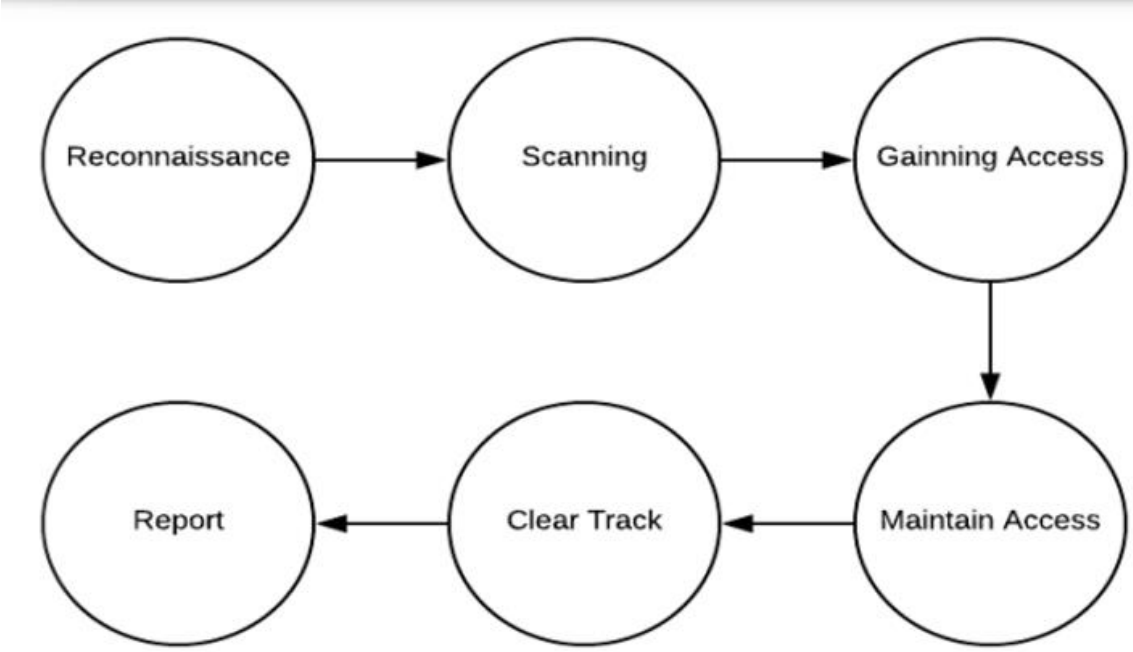


Figure 4: Activity diagram for recon method

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent. In both cases an activity diagram will have a beginning (an initial state) and an end (a final state).

The goal of recon is to gather as much information about the target as you can. More the information, more beneficial it will be for further phases of pen testing. Most of new learners underestimates this phase and ignore it but recon is most important phase of pen testing. Your point of view for digital world changes if you cFompletely understood this process. Learning to successfully conduct the recon process is a valuable skill for anyone.

Much like car burglars test door handles to see which cars are locked, a port scan is a process which identifies “open doors” to a computer. Ports are points at which information comes and goes from a computer, so by scanning for open ports, attackers can find weakened pathways with which to enter your computer.

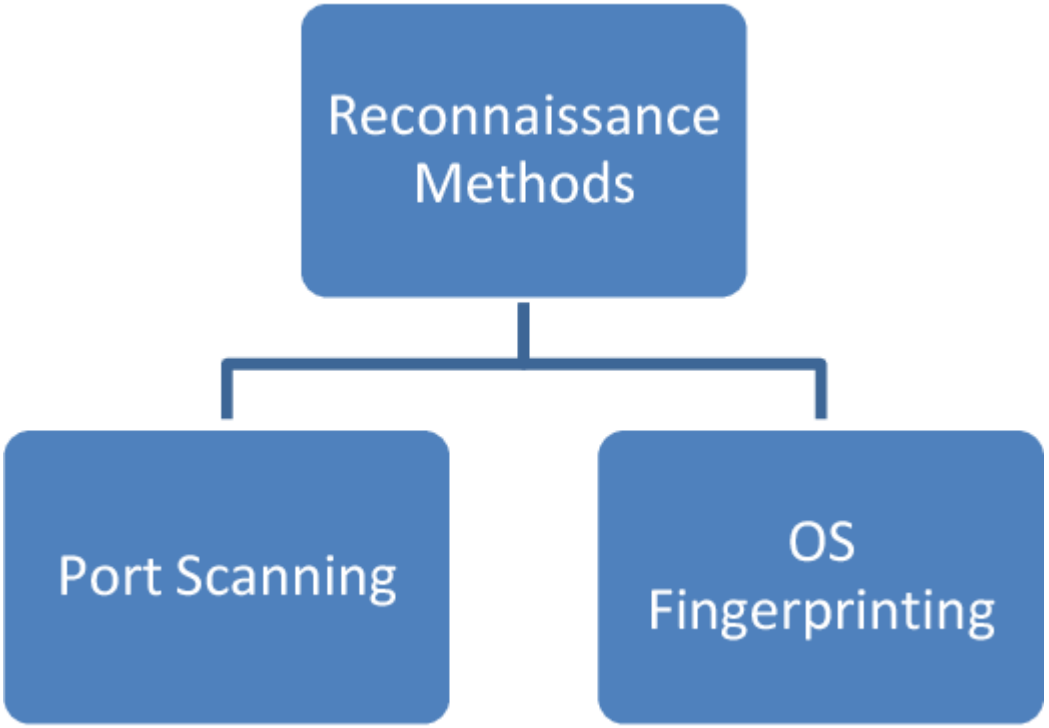


Figure 5: Types of recon method

## 4.4 SEQUENCEDIAGRAM

In the sequence diagram how, the object interacts with the other object is shown. There is sequence of events that are represented by a sequence diagram. It is a time-oriented view of the interaction between objects to accomplish a behavioral goal of the system. Figure, represents the job sequence of police and Figure, and represents the job sequence of the admin.

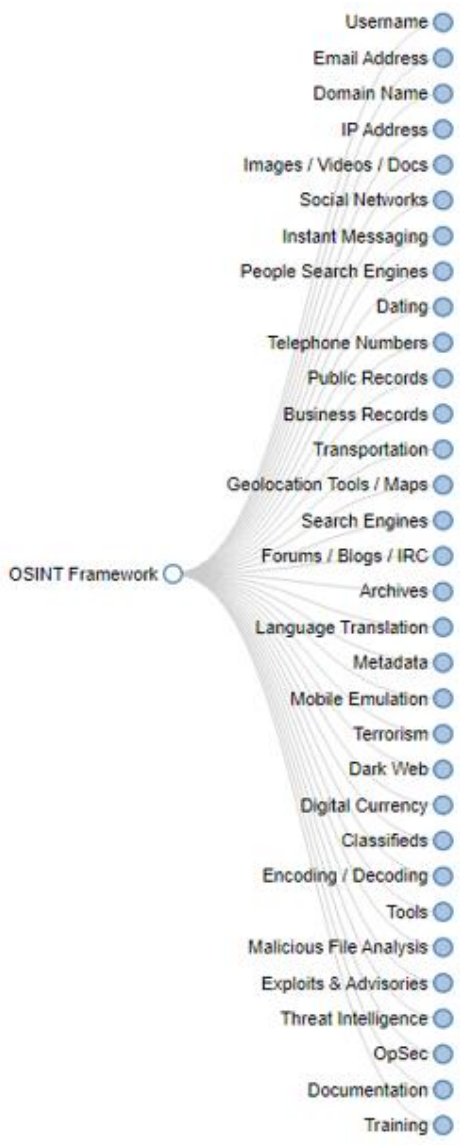


Figure 6: Sequence diagram for recon method.

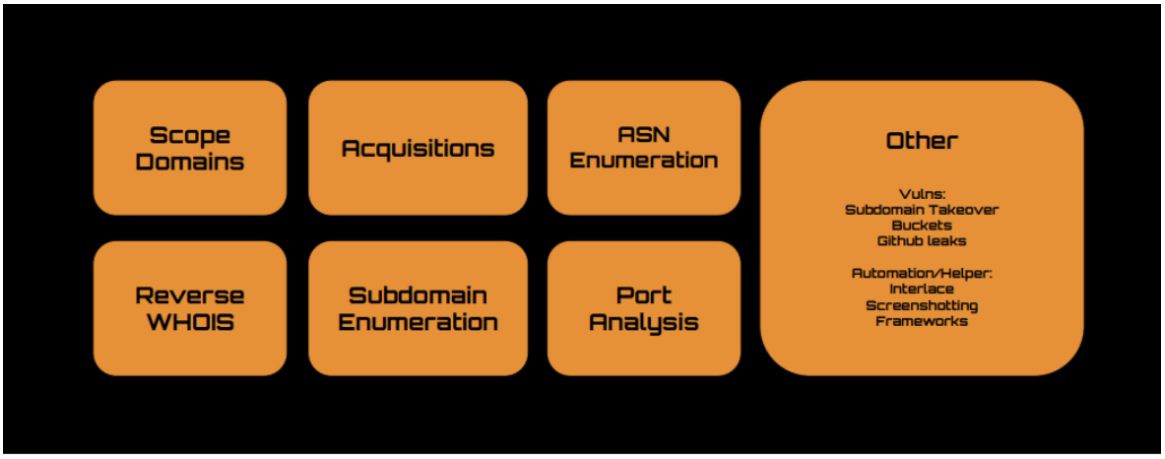


Figure 7: recon method

## 4.5 UML DIAGRAM

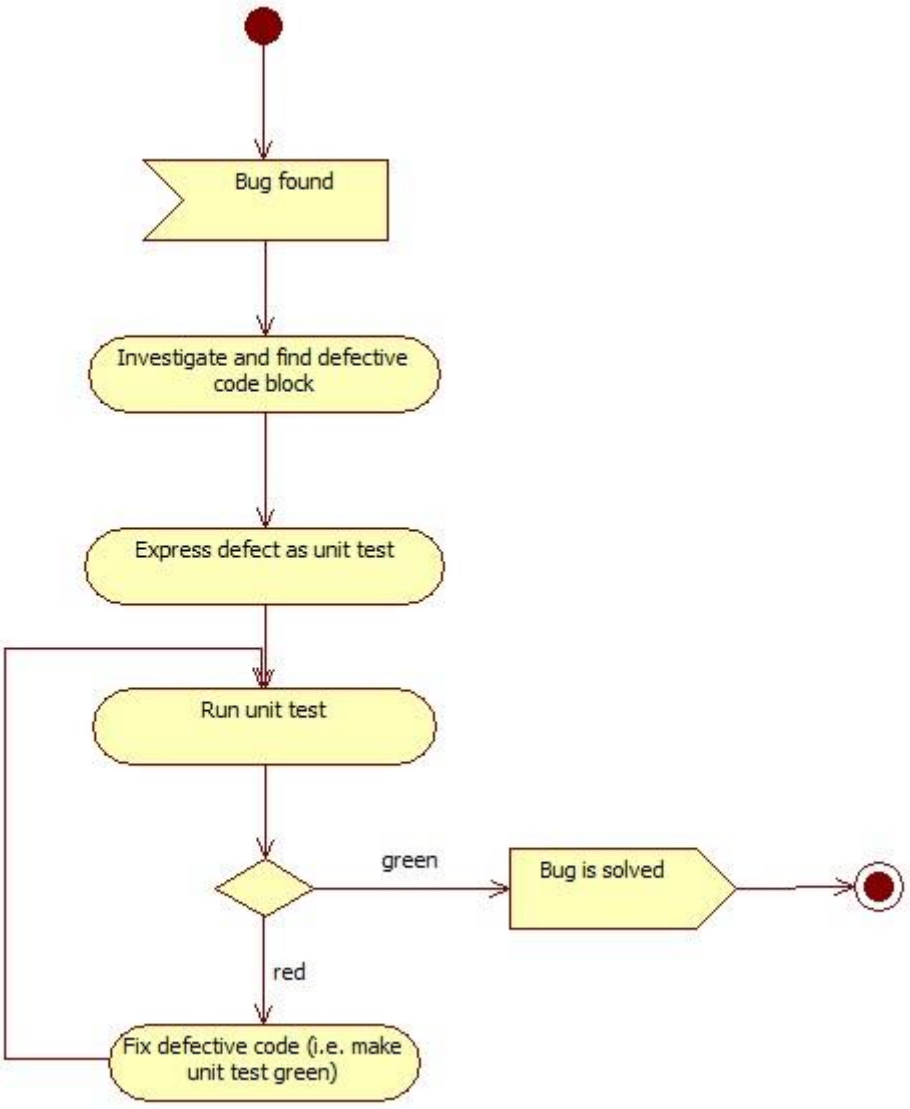


Figure 8: UML Diagram for bug bounty

# 5. IMPLEMENTATION

The implementation of the project is done with the help of python language. To be particular, for the purpose of machine learning Anaconda is being used. Anaconda is one of several Python distributions. Anaconda is a new distribution of the Python. It was formerly known as Continuum Analytics. Anaconda has more than 100 new packages. Anaconda is used for scientific computing, data science, statistical analysis, and machine learning. On Python technology, we found out Anaconda to be easier.

Since it helps with the following problems:

- Installing Python on multiple platforms.
- Separating out different environments.
- Dealing with not having correct privileges.
- Getting up and running with specific pack ages and libraries.

This data was scraped from the publicly available data from Indore police website which had been made by people in police station of different areas. Implementation of the idea started from the Indore city itself so as to limit an area for the prediction and making it less complex. The data was sorted and converted into a new format of time stamp, longitude, latitude, which was the input that machine would be taking so as to predict the crime rate in particular location or city.

## 5.1 ENVIRONMENTAL SETUP

Subfinder is a subdomain discovery tool that discovers valid subdomains for websites by using passive online sources. It has a simple modular architecture and is optimized for speed. subfinder is built for doing one thing only - passive subdomain enumeration, and it does that very well.

We have designed subfinder to comply with all passive source’s licenses, and usage restrictions, as well as maintained a consistently passive model to make it useful to both penetration testers and bug bounty hunters alike.

Subfinder will work after using the installation instructions however to configure Subfinder to work with certain services, you will need to have setup API keys.

These values are stored in the $HOME/.config/subfinder/provider-config.yaml file which will be created when you run the tool for the first time. The configuration file uses the YAML format. Multiple API keys can be specified for each of these services from which one of them will be used for enumeration.

Sublist3r is a python tool designed to enumerate subdomains of websites using OSINT. It helps penetration testers and bug hunters collect and gather subdomains for the domain they are targeting. Sublist3r enumerates subdomains using many search engines such as Google, Yahoo, Bing, Baidu and Ask. Sublist3r also enumerates subdomains using Netcraft, Virustotal, ThreatCrowd, DNSdumpster and ReverseDNS.

Sub brute was integrated with Sublist3r to increase the possibility of finding more subdomains using bruteforce with an improved wordlist. The credit goes to TheRook who is the author of subbrute.

Httpx is a fast and multi-purpose HTTP toolkit allow to run multiple probers using library, it is designed to maintain the result reliability with increased threads.

This will run the tool against all the hosts and subdomains in hosts.txt and returns URLs running HTTP webserver. Aquatone for your operating system.

Uncompress the zip file and move the aquatone binary to your desired location.You probably want to move it to a location in your $PATH for easier use. If you for some reason don't trust the pre-compiled binaries, you can also compile the code yourself. You are on your own if you want to do this. I do not support compiling problems.

Aquatone is a set of tools used for performing reconnaissance, scanning, and discovery domain names. Aquatone can discover subdomains on a given target domain using OSINT source and the most common domain brute force method. After discovering the subdomain, the Aquatone tool can scan the domain for standard web ports and HTTP headers information. HTML bodies and snapshots can be collected and considered as the report to analyze the attack environment quickly.

## 5.2 MODULE DESCRIPTION

Bug bounty programs benefit companies by making use of hackers who can uncover the bugs in the companies' codes. These programs have access to a larger number of hackers or testers, thereby increasing the chances of finding bugs before malicious hackers attempt to exploit them.

A bug bounty program, also known as a vulnerability rewards program (VRP), offering offers rewards to individuals for uncovering and reporting software bugs. As part of a vulnerability management strategy, these crowdsourcing initiatives are often used by companies to supplement penetration tests and internal code audits.

It can serve as a good public relations choice for companies. These programs can also serve as an indication to the public and regulators that a company has a mature security program.

The popularity of these programs is likely to continue, as they have come to be considered an industry standard that should be invested in by all companies.

Bug bounty programs benefit companies by making use of hackers who can uncover the bugs in the companies' codes. These programs have access to a larger number of hackers or testers, thereby increasing the chances of finding bugs before malicious hackers attempt to exploit them.

It can serve as a good public relations choice for companies. These programs can also serve as an indication to the public and regulators that a company has a mature security program.

Bug bounty programs have become increasingly prominent in the public and private sector due to the various benefits offered by them to the company that is being tested.

The key benefit of a bug bounty program is that the company hosting it can have a number of vulnerabilities within its applications found and fixed, thus preventing exploitation by cybercriminals and preventing significant damage.

The program provides a higher probability of finding vulnerabilities, helping to protect the company's reputation, and decreasing high-value hacks.

Bug bounty programs enable significant cost savings in several ways. Firstly, paying a bounty to learn about a vulnerability cost much less than attempting to

remediate a cyber security incident due to that same vulnerability. While bounty values are subject to variation, even the most expensive bounties are often significantly cheaper than data breaches.

Because companies have to pay the bug bounty hunters only if they find something, bug bounty programs are, ultimately, much cheaper than paying for the same level of security testing via contractors as they have to be paid by the hour whether or not they find anything.

## 5.3 SOFTWARE DESCRIPTION

Go (also called Golang or Go language) is an open-source programming language used for general purpose. Go was developed by Google engineers to create dependable and efficient software. Most similarly modelled after C, Go is statically typed and explicit.

The language was designed by taking inspiration for the productivity and relative simplicity of Python, with the ability of C. Some of the problems that Go addresses are slow build time, uncontrolled dependencies, effort duplication, difficulty of writing automatic tools and cross-language development.

Go works by using "go routines," or lightweight processes, which allows further efficiencies. Go also uses a collection of packages for efficient dependency management.

Go includes a number of features such as its standard library, package management, static typing, support for testing as well as its platform independence. Go's standard library is based off the use of distributed packages. Package management refers to how Go will manage support for user-based and external package management. Packages can be published using a small set of commands. Static typing is a type system that ensures conversions and compatibility while avoiding the issues that come with dynamically typed languages. Go also supports unit tests to run in parallel with written code. In addition, due to Go's modular design, the code can be compiled onto almost any platform.

More specifically, Go uses lightweight processes that enable concurrent processing and behave like threads. The syntax will mimic patterns commonly seen in dynamic languages. Golang favors composition interfaces over inheritance. Some

of Go's tools worth highlighting are its "Gofmt" feature that automatically formats and indents code for readability, "Go run" that compiles and runs code simultaneously, "Go get" that seamlessly integrates with GitHub and "Godoc" that generates HTML-based documentation according to the code structure and developer comments.

Analysis software tools should have the ability to use historical RMS data to produce targeted crime forecasts that are accurate. In any agency, utilizing crime analysis software in either a simple or more robust application should lead to more efficiently allocated Officer Resources and help to improve the focus of their time and efforts on proactively reducing and preventing crime.

Golang has comparable features to other programming languages but provides an overall unique alternative. The main design goal of Go is to facilitate fast compilation, unlike some of the other common languages. When compared to C++, Go reduces the amount of runtime errors and dependencies while increasing memory safety and memory management.

Sublist3r is a python tool designed to enumerate subdomains of websites using OSINT. It helps penetration testers and bug hunters collect and gather subdomains for the domain they are targeting. Sublist3r enumerates subdomains using many search engines such as Google, Yahoo, Bing, Baidu and Ask. Sublist3r also enumerates subdomains using Netcraft, Virustotal, ThreatCrowd, DNSdumpster and ReverseDNS.

Sub brute was integrated with Sublist3r to increase the possibility of finding more subdomains using bruteforce with an improved wordlist. The credit goes to TheRook who is the author of subbrute.

Httpx is a fast and multi-purpose HTTP toolkit allow to run multiple probers using library, it is designed to maintain the result reliability with increased threads.

This will run the tool against all the hosts and subdomains in hosts.txt and returns URLs running HTTP webserver. Aquatone for your operating system.

Uncompress the zip file and move the aquatone binary to your desired location.You probably want to move it to a location in your $PATH for easier use. If you for some reason don't trust the pre-compiled binaries, you can also compile the code yourself. You are on your own if you want to do this. I do not support compiling problems.

Aquatone is a set of tools used for performing reconnaissance, scanning, and discovery domain names. Aquatone can discover subdomains on a given target domain using OSINT source and the most common domain brute force method. After discovering the subdomain, the Aquatone tool can scan the domain for standard web ports and HTTP headers information. HTML bodies and snapshots can be collected and considered as the report to analyze the attack environment quickly.

## 5.4 SAMPLE CODES

### First Script:

```
#!/bin/bash

#getting linux update
apt-get update
#installing design tools
apt-get install figlet toilet
cd
cd Downloads

#install go language
wget https://go.dev/dl/go1.18.3.linux-amd64.tar.gz
tar -C /usr/local/ -xzf go1.18.3.linux-amd64.tar.gz

#setting go path environmental variable
export GOPATH=/usr/local
export GOROOT=/usr/local/go
PATH=$PATH:$GOROOT/bin:$GOPATH/bin
echo 'export GOPATH=/usr/local' >> ~/.zshrc
echo 'export GOROOT=/usr/local/go' >> ~/.zshrc
echo 'PATH=$PATH:$GOROOT/bin:$GOPATH/bin' >> ~/.zshrc
figlet go installation completed

echo "run: source ~/.zshrc"
```

### Second Script:

```
#!/bin/bash

figlet installing subdomain finder
#installing subdomains finder tool
go install -v github.com/projectdiscovery/subfinder/v2/cmd/subfinder@latest
figlet installing service finder tool
#installing service finder tool
go install -v github.com/projectdiscovery/httpx/cmd/httpx@latest
figlet installing spidering tool
#installing gospider tool
GO111MODULE=on go install github.com/jaeles-project/gospider@latest
figlet installation screentshotting tool
#installing screentshotting tool
wget
https://github.com/michenriksen/aquatone/releases/download/v1.7.0/aquatone_linux_amd
64_1.7.0.zip
```

#unzipping the screenshotting tool
unzip aquatone_linux_amd64_1.7.0.zip -d /usr/local/bin

figlet installation completed

## Third Script:

```
#!/bin/bash


NC='\033[0m'
RED='\033[1;38;5;196m'
GREEN='\033[1;38;5;040m'
ORANGE='\033[1;38;5;202m'
BLUE='\033[1;38;5;012m'
BLUE2='\033[1;38;5;032m'
PINK='\033[1;38;5;013m'
GRAY='\033[1;38;5;004m'
NEW='\033[1;38;5;154m'
YELLOW='\033[1;38;5;214m'
CG='\033[1;38;5;087m'
CP='\033[1;38;5;221m'
CPO='\033[1;38;5;205m'
CN='\033[1;38;5;247m'
CNC='\033[1;38;5;051m'

echo -e ${RED}
"############################################################"
echo -e ${CP}    "#  ____    _    ____    ____  _____ ____ ___  _   _     #"
echo -e ${CP}    "# / ___|  / \  |  _ \  |  _ \| ____/ ___/ _ \| \ | |    #"
echo -e ${CP}    "# \___ \ / _ \ | |_) | | |_) |  _|| |  | | | |  \| |    #"
echo -e ${CP}    "#  ___) / ___ \|  _ <  |  _ <| |__| |__| |_| | |\  |    #"
echo -e ${CP}    "# |____/_/   \_\_| \_\ |_| \_\_____\____\___/|_| \_|    #"
echo -e ${CP}    "#              Automate your Recon                       #"
echo -e ${BLUE}  "#            ARYAN                                 #"
echo -e ${YELLOW} "#                          SOWMYA                        #"
echo -e ${CG}    "#                          RAKESH                      #"
echo -e ${RED}
"############################################################"
sleep 2
d=$(date +"%b-%d-%y %H:%M")

echo -n -e ${ORANGE}"
 ____ _____  _    ____ _____ ___ _   _  ____
/ ___|_   _|/ \  |  _ \_   _|_ _| \ | |/ ___|
\___ \ | | / _ \ | |_) || |  | ||  \| | |  _
 ___) || |/ ___ \|  _ < | |  | || |\  | |_| |
|____/ |_/_/   \_\_| \_\|_| |___|_| \_|\____|
```

```
 _____  ______ _____ ____  _   _ _   _    _    ____ _____    _    _   _ _____ ______
|  _ \| _____/ ____/ _ \| \ | | | / \  |_ _/ ____|  / \  | \ | |/ ____| _____|
| |_) |  _|| |  | | | |  \| |  \| | / _ \  | |\____ \ / _ \ |  \| | |   |  _|
|  _ <| |__| |__| |_| | |\  | |\  |/ ____ \ | | ____) / ____ \| |\  | |___| |___
|_| \_\_____\_____\____/|_| \_|_| \_/_/   \_\_\____|____/_/   \_\_| \_|\_____|_____|
\n[+] Enter domain (e.g evil.com) : "

#reading domain namef
  read domain
mkdir -p /root/recon/$domain

echo -e ${BLUE}"\n[+] Recon Started On $d: \n"
figlet FINDING SUBDOMAINS
#finding subdomains for main domains
subfinder -d $domain -silent -o /root/recon/$domain/subdomain.txt
sleep 1


figlet FINDING SERVICES OF SUBDOMAINS
#finding services of subdomains
httpx -l /root/recon/$domain/subdomain.txt -ip -sc -ct -td -probe -o
/root/recon/$domain/httpx.txt
sleep 1


echo -e ${BLUE2}"[+]\n"
figlet WORKING DOMAINS
#fetching working domains
cat /root/recon/$domain/httpx.txt | grep -e "200" -e "301" -e "302" | awk '{print$1}' | tee
/root/recon/$domain/200ok.txt
sleep 1

echo -e ${RED}"\n[+]\n"
figlet FAILED DOMAINS
#fetching failed domains
cat /root/recon/$domain/httpx.txt | grep -e "FAILED" -e "401" -e "402" -e "403" -e "404"
-e "502" -e "503" | awk '{print$1}' | tee /root/recon/$domain/500fail.txt
sleep 1


echo -e ${CPO}"\n[+]\n"
figlet FETCHING IP ADDRESS
#fetching ip address of active domains
cat /root/recon/$domain/httpx.txt | awk '{print$5}' | tee /root/recon/$domain/ip.txt
sleep 1


echo -e ${GREEN}"[+]\n"
figlet CAPTURING WORKING DOMAINS
```

```
#making screenshots of passed subdoamins
cat /root/recon/$domain/200ok.txt | aquatone -out /root/recon/$domain/
cd recon/$domain/
mv screenshots 200ok
rm aquatone_report.html
rm aquatone_session.json
rm aquatone_urls.txt
rm -rf headers
rm -rf html
cd
sleep 1


echo -e ${RED}" \n[+]\n"
figlet CAPTURING NON-WORKING DOMAINS
#making screenshots of failed subdoamins
cat /root/recon/$domain/500fail.txt | aquatone -out /root/recon/$domain/
cd recon/$domain/
mv screenshots 500fail
rm aquatone_report.html
rm aquatone_session.json
rm aquatone_urls.txt
rm -rf headers
rm -rf html
cd
sleep 1


echo -e ${BLUE2}"\n[+]\n"
figlet spidering domains
#running gospider tool in order to spider the subdomains
gospider -S /root/recon/$domain/200ok.txt | grep "$domain" | tee
/root/recon/$domain/spider.txt

echo -e ${ORANGE}"
 _____  ______ _____ ____  _   _ _   _    _    ____ _____    _    _   _ _____ ______
|  _ \| _____/ ____/ _ \| \ | | | / \  |_ _/ ____|  / \  | \ | |/ ____| _____|
| |_) |  _|| |  | | | |  \| |  \| | / _ \  | |\____ \ / _ \ |  \| | |   |  _|
|  _ <| |___| |___| |_| | |\  |  |/ ____ \ | | ____) / ____ \| |\  | |___| |___
|_| \_\______\_____\____/|_| \_|_| \_/_/   \_\_\____|_____/_/   \_\_| \_|\_____|______|


  _____ ____   __  __ _____  _     _____ _____ _____ _____
 / ____/ _ \|  \/  |  _ \| |   | _____|_   _| _____|  _ \
| |  | | | | |\/| | |_) | |   |  _|   | | |  _| | | | |
| |__| |_| | |  | |  __/| |___| |___  | | | |___| |_| |
 \_____\___/|_|  |_|_|   |_____|_____| |_| |_____|____/
                                                    "
```

## 5.5 INSTALLATION SCREENS:

## Updating Linux:

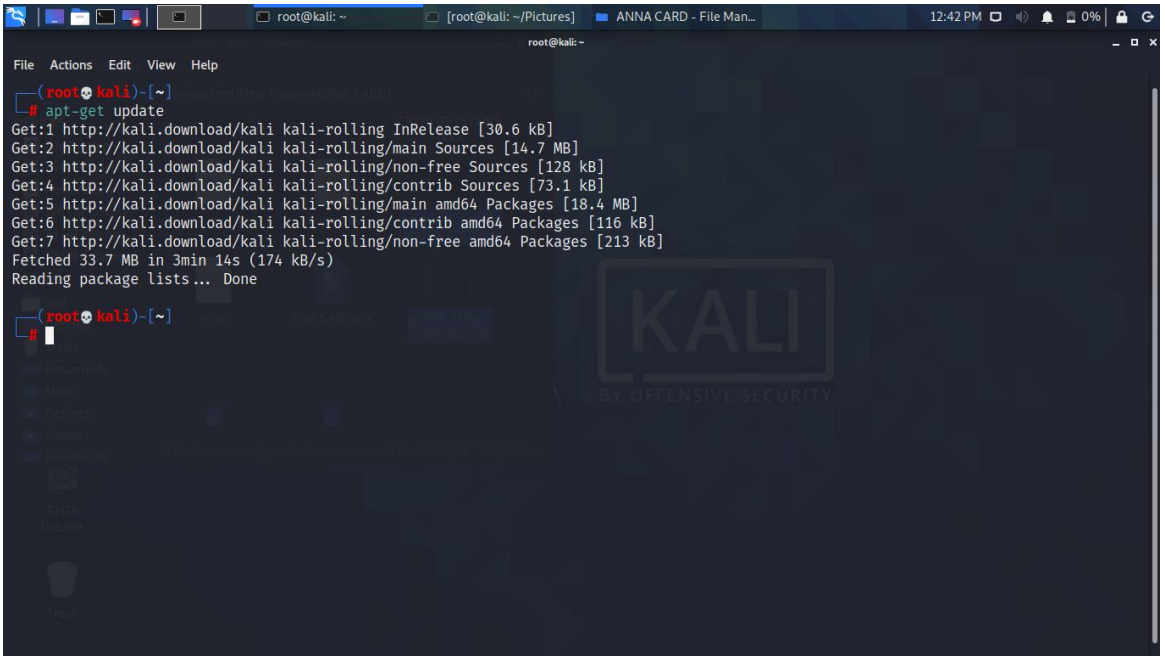


Figure 1: getting system-update.

## Installing Python

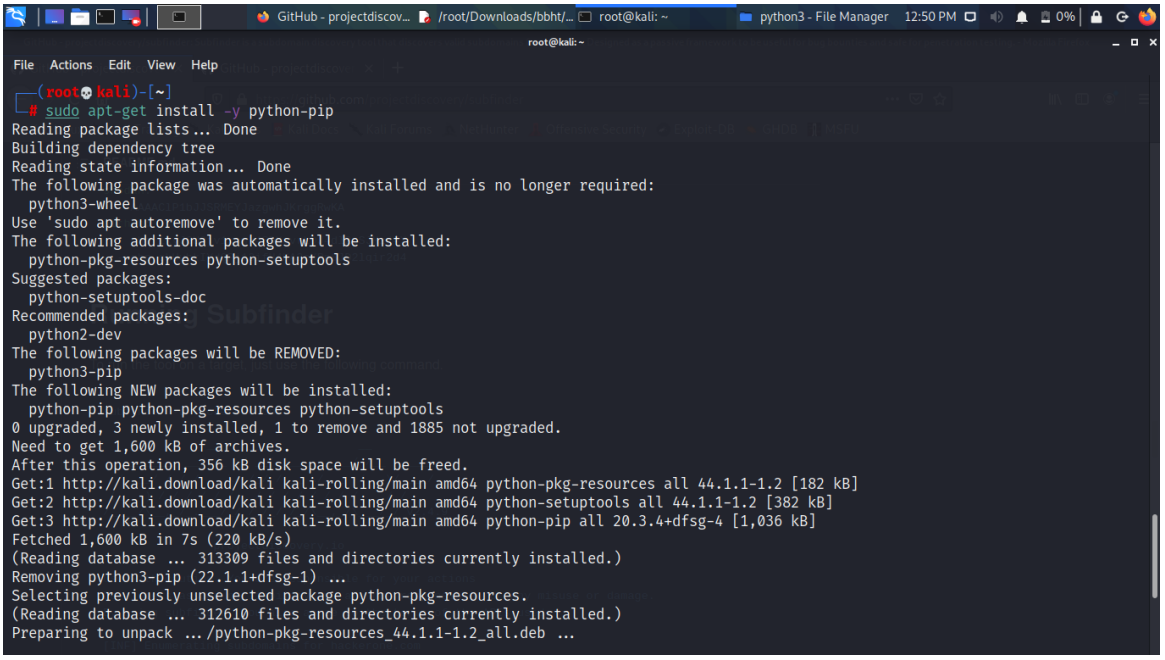


Figure 2: installing python.

## Installing pip:

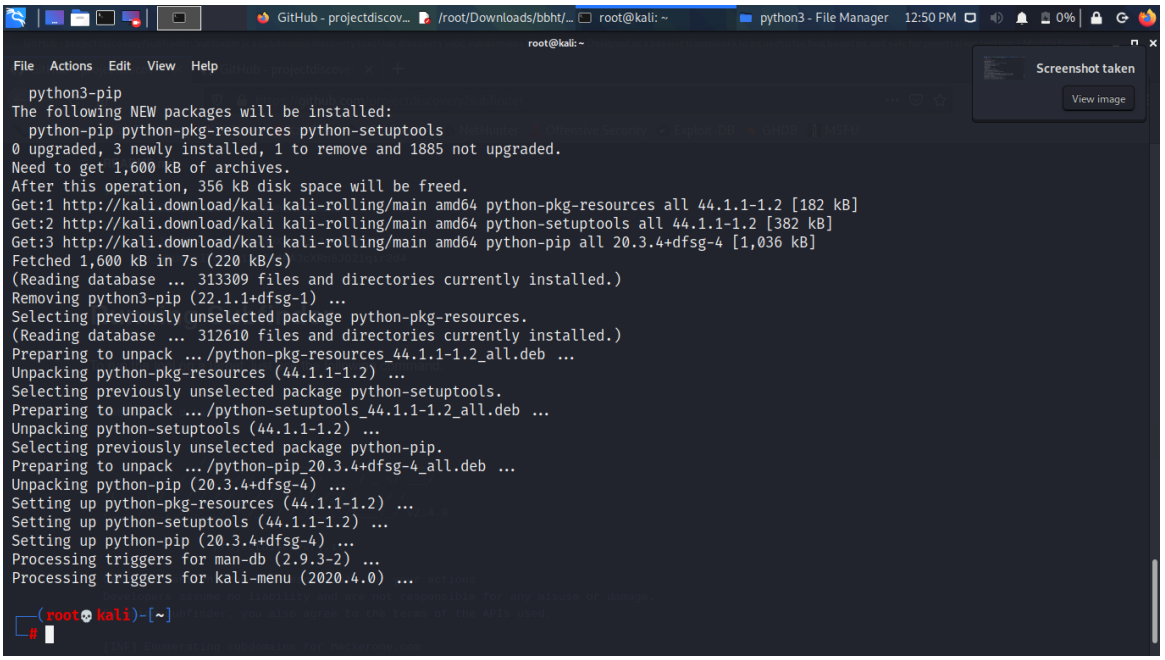


Figure 3: installing pip.

## Installing Python3:

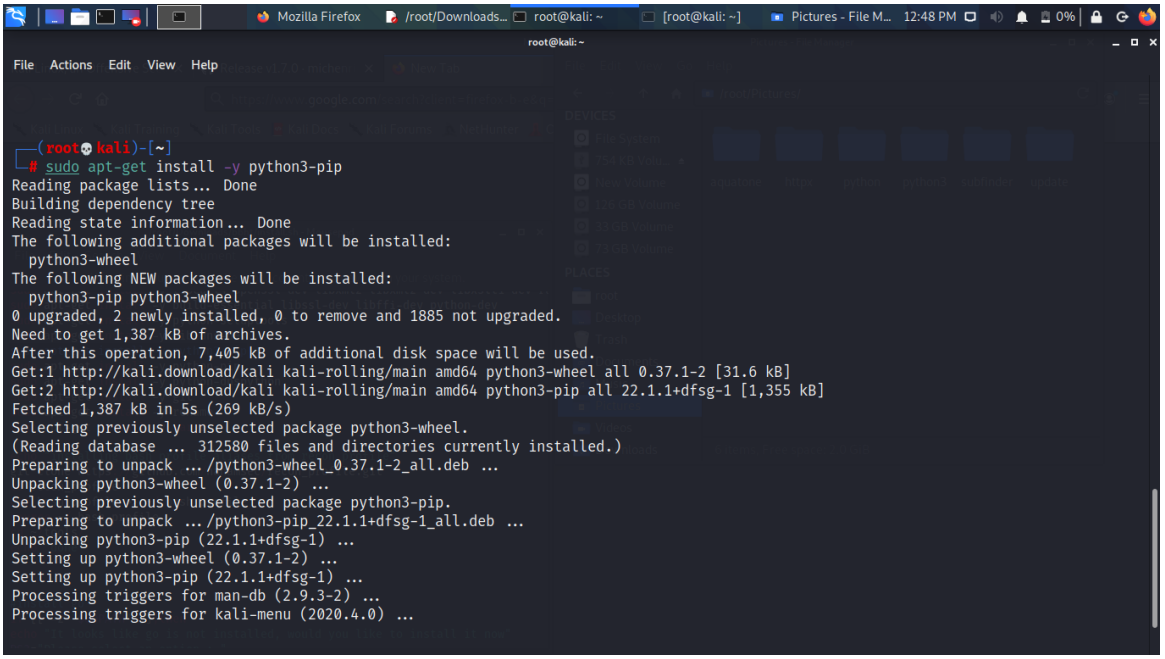


Figure 4: installing python3.

## Installing Subdomain Finding Tool:

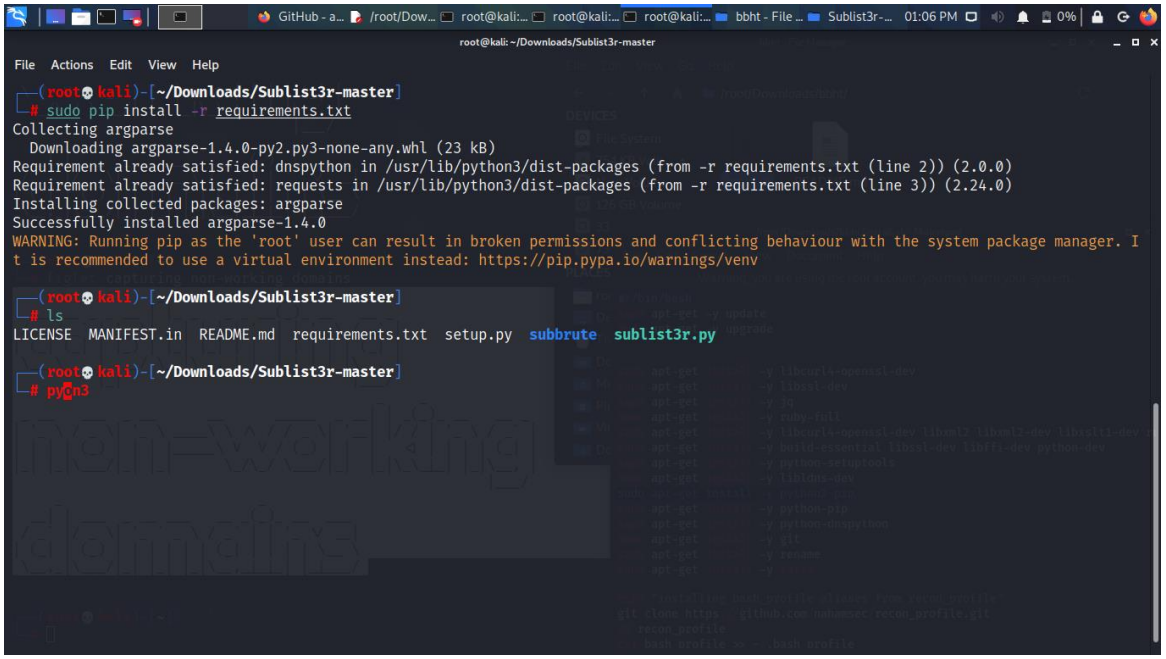


Figure 5: installing subdomain finding tool.

## Using Subdomain Finding Tool:

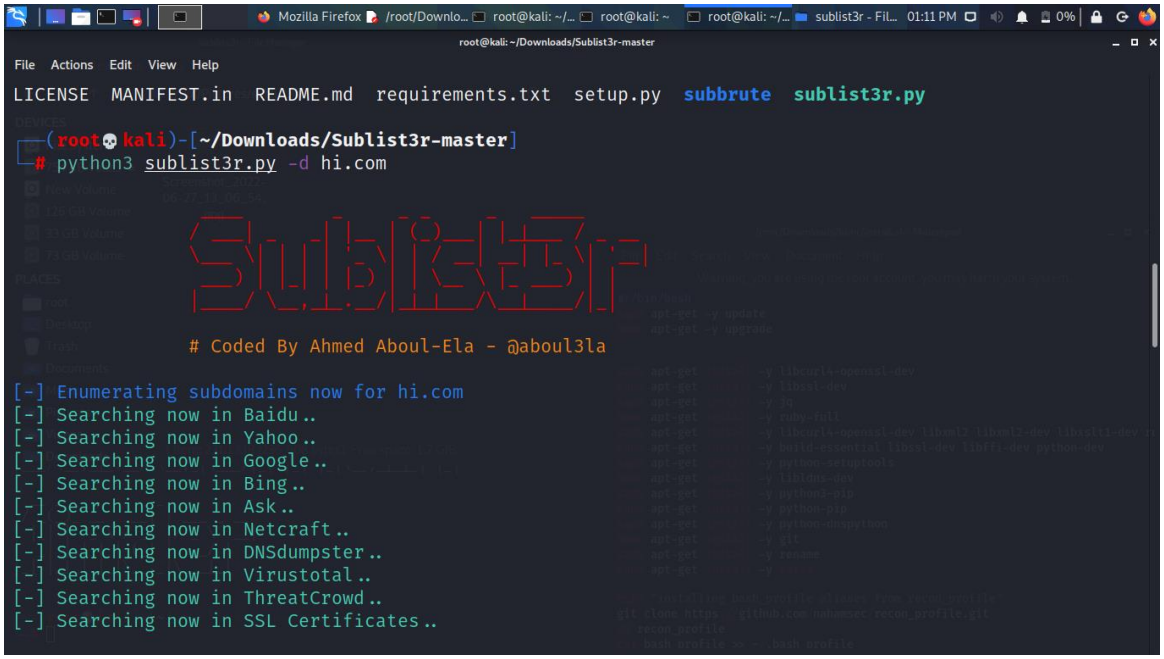


Figure 6: using subdomain finding tool.

## Installing GO Language:

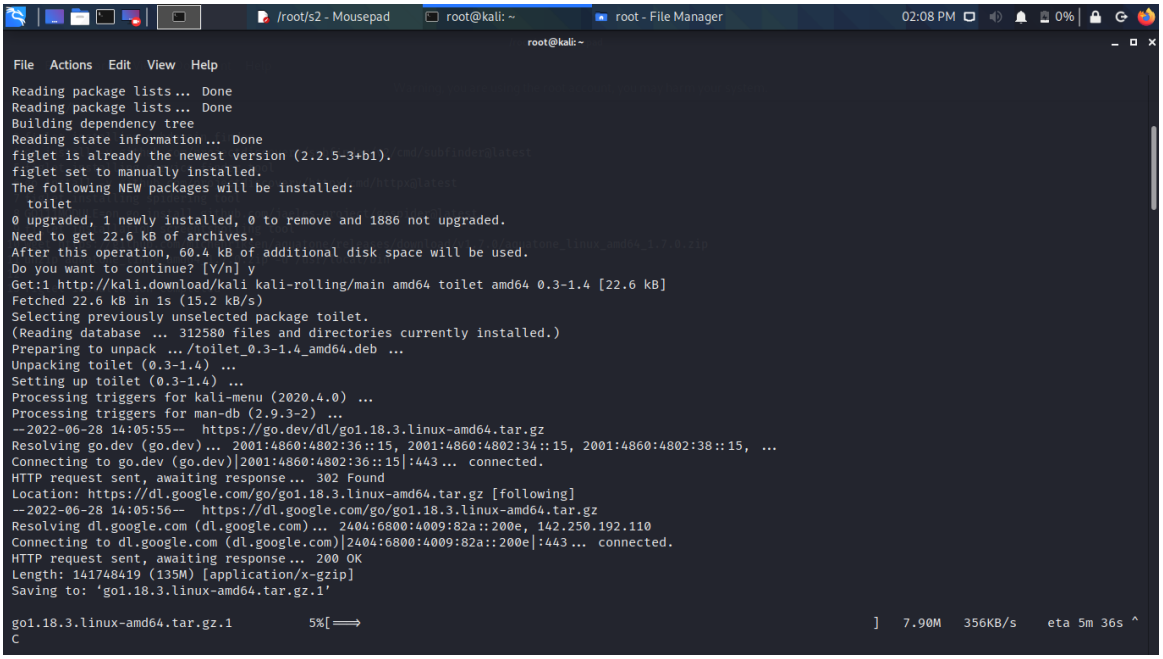


Figure 7: installing main go language.

## Installing Second-Subdomain Finder Tool:


Figure 8: installing second-subdomain finding tool.

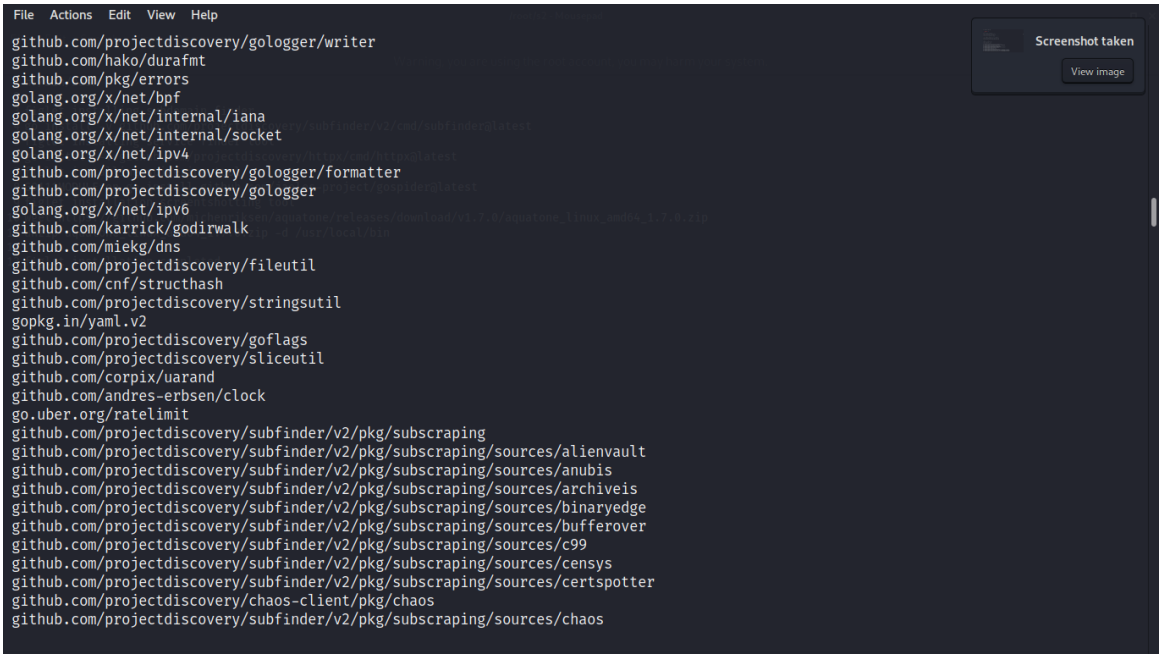


Figure 9: installing tool extensions.

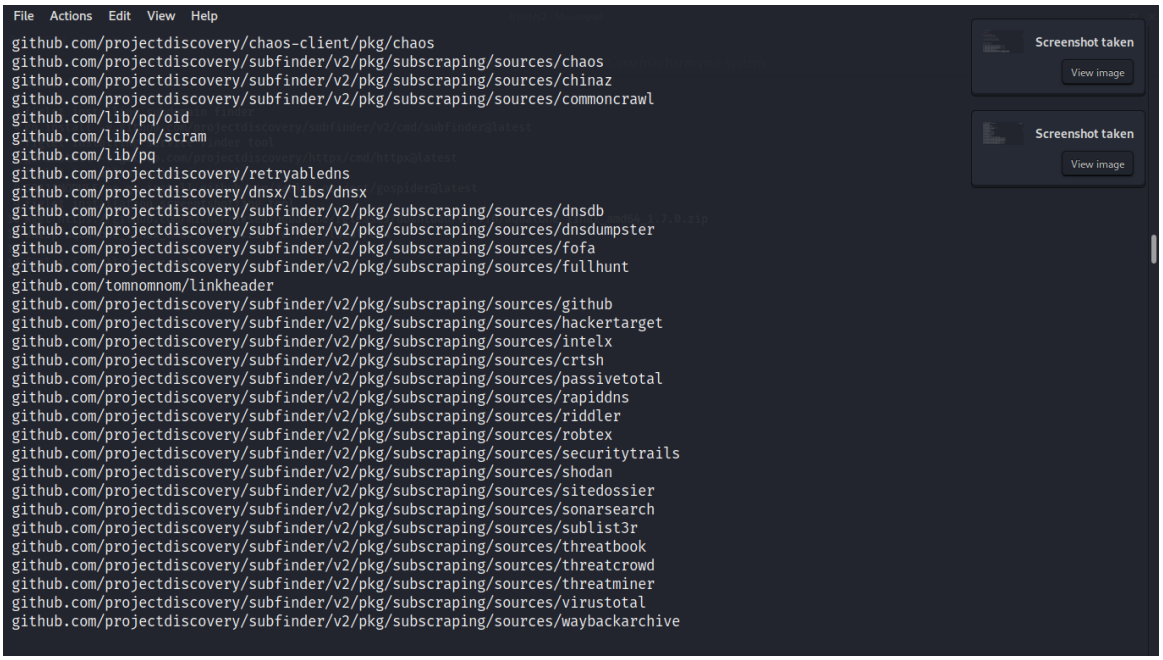


Figure 10: installing tool extensions.

## Installing Service Finding Tool:

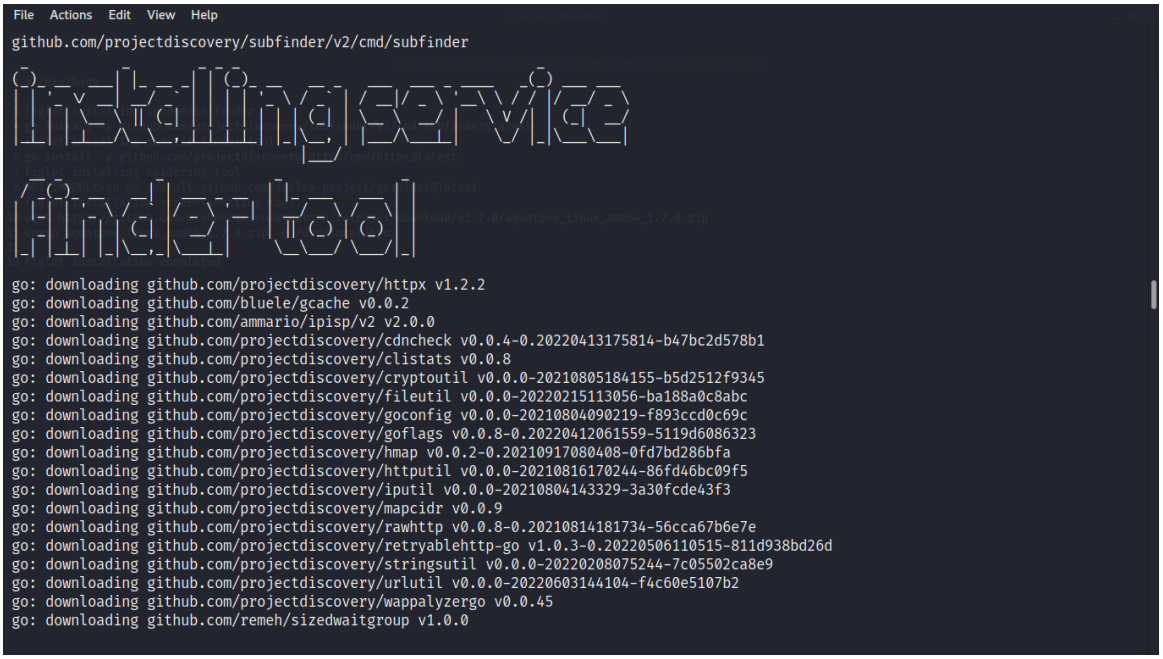


Figure 11: installing tool extensions.

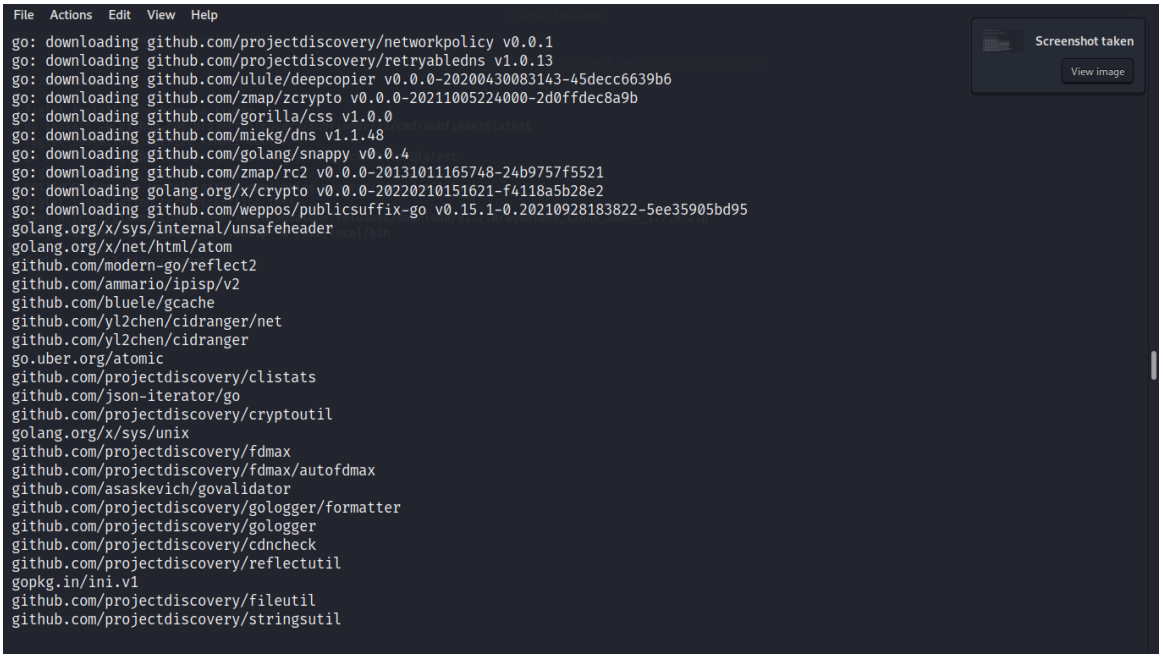


Figure 12: installing tool extensions .

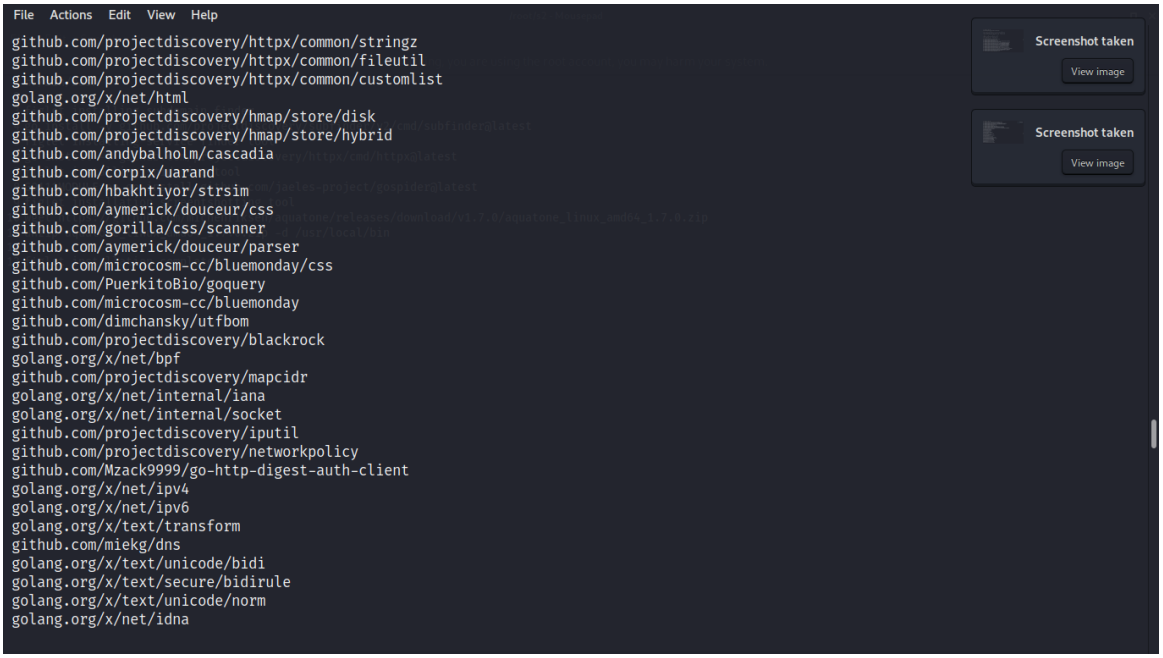


Figure 13: installing tool extensions.

## Installing Spidering tool

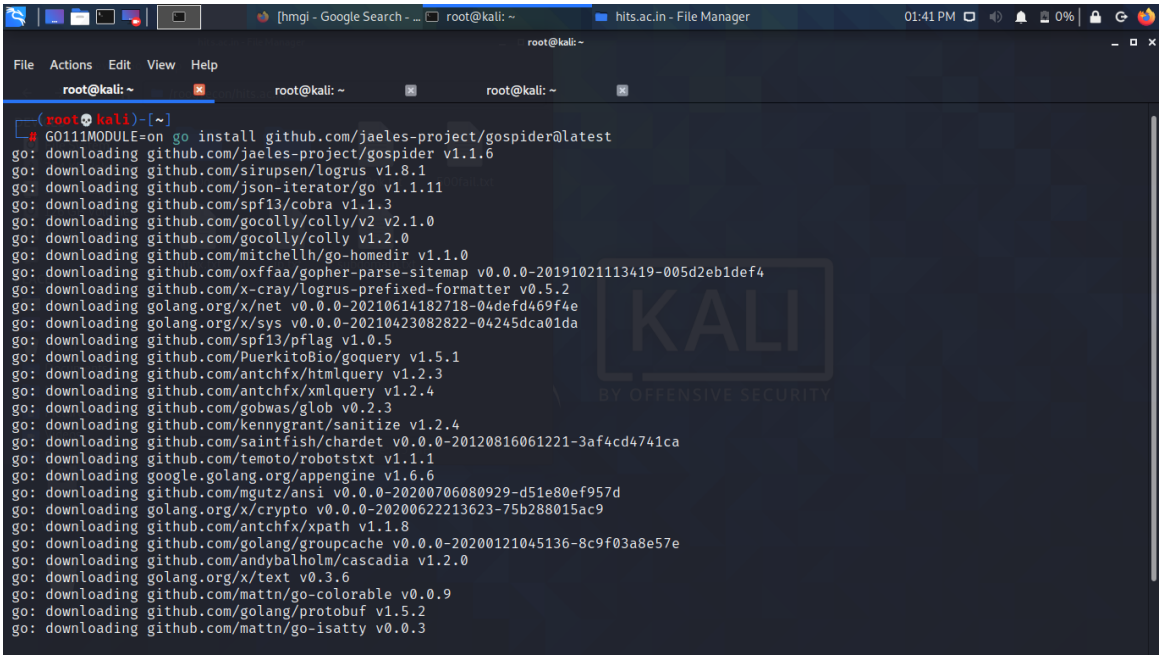


Figure 14: installing spidering tool

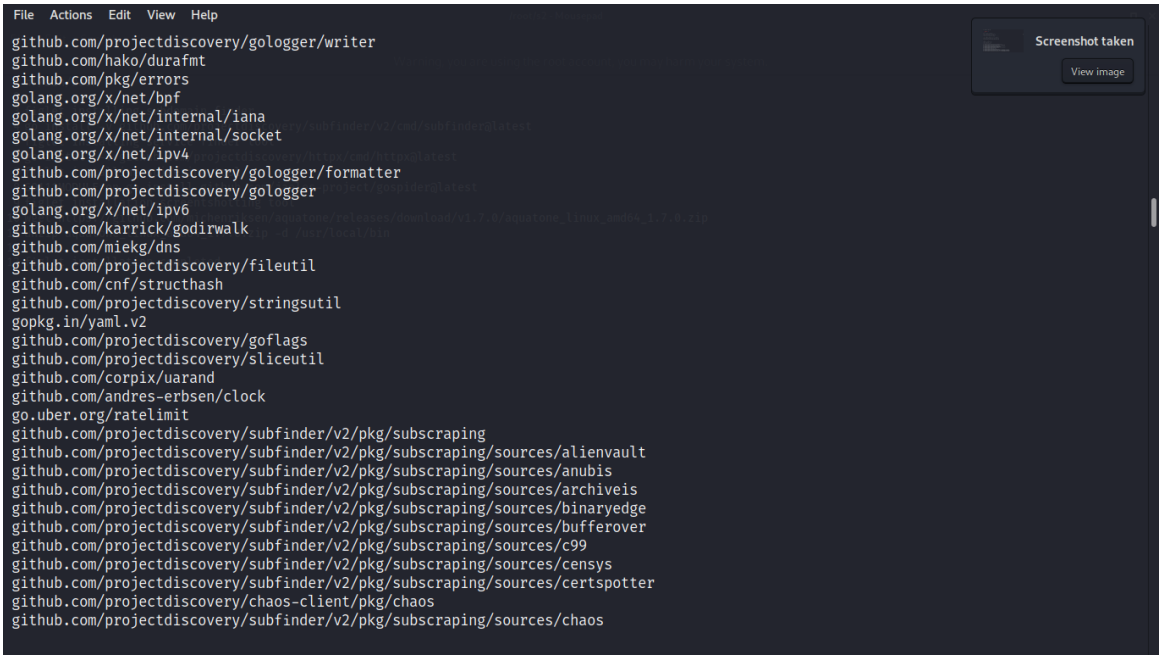


Figure 15: installing spidering tool

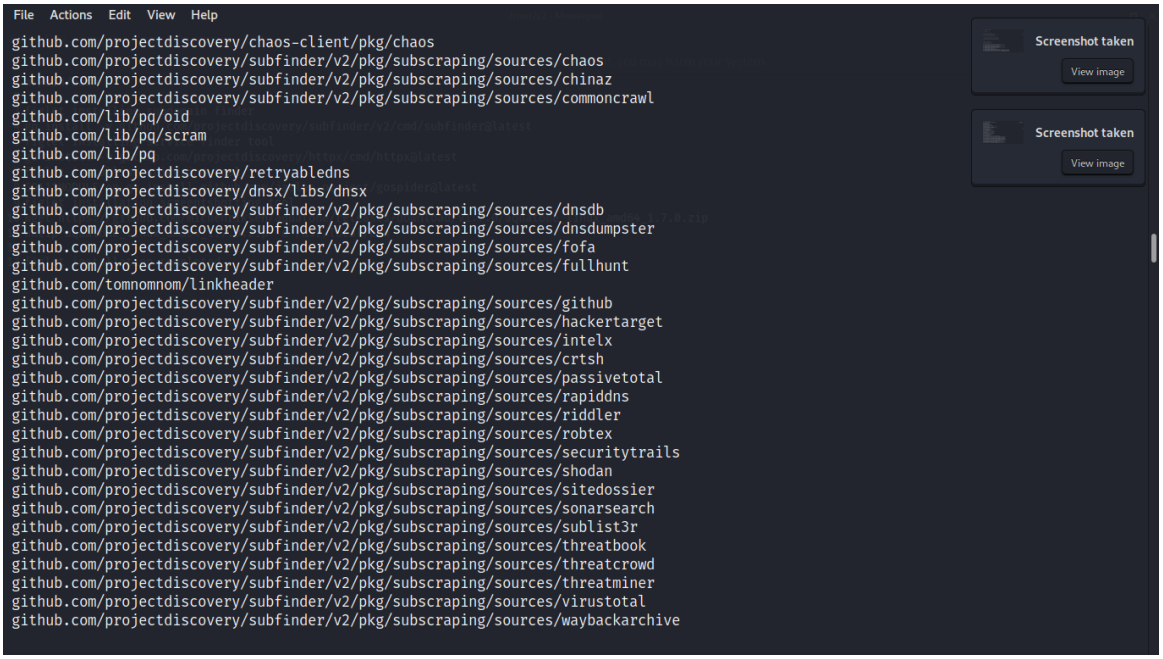


Figure 16: installing spidering tool

## Installing Screenshotting tool:

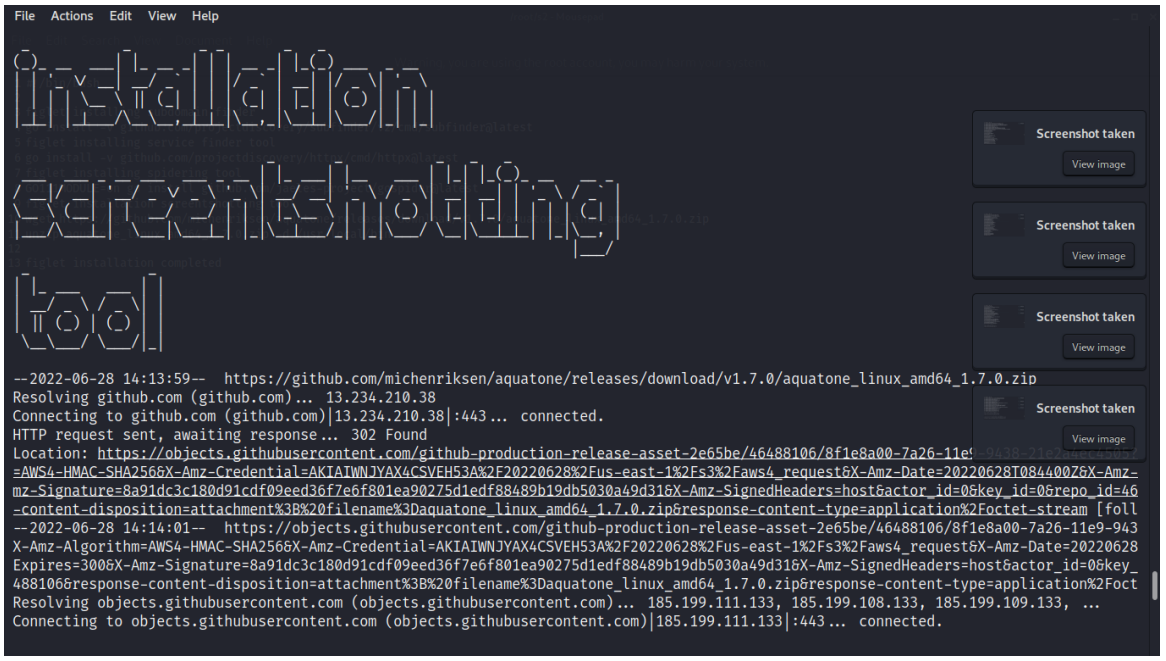


Figure 17: installing screenshotting tool.

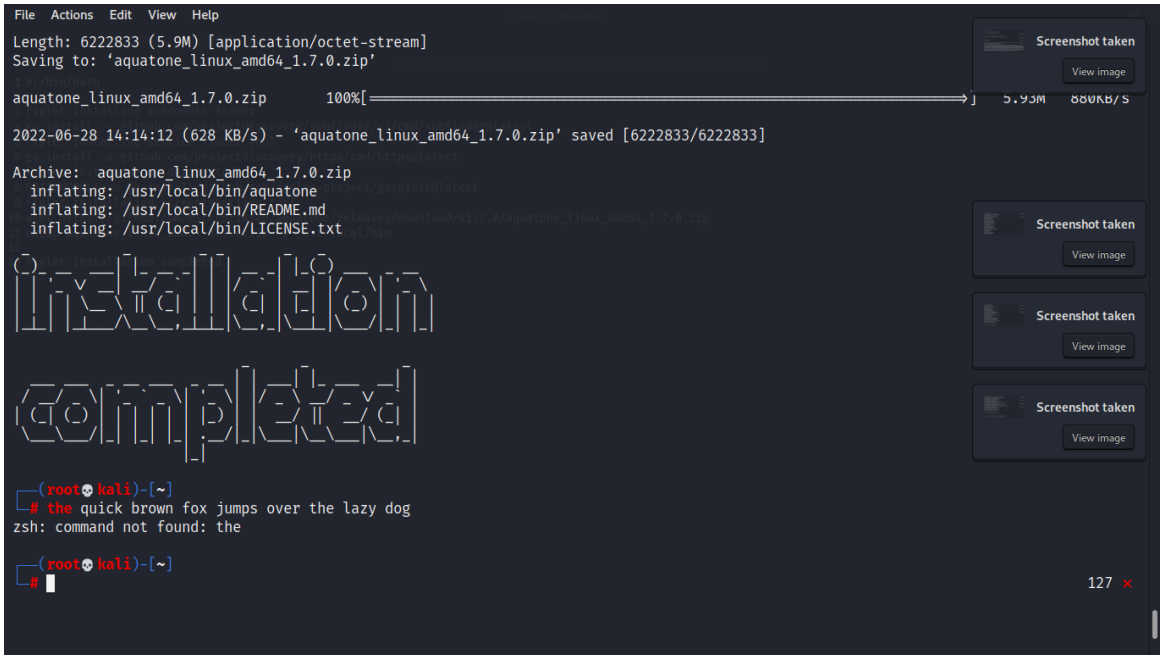


Figure 18: path setting and installation completed screen.

## OUPTPUT SCREENS:

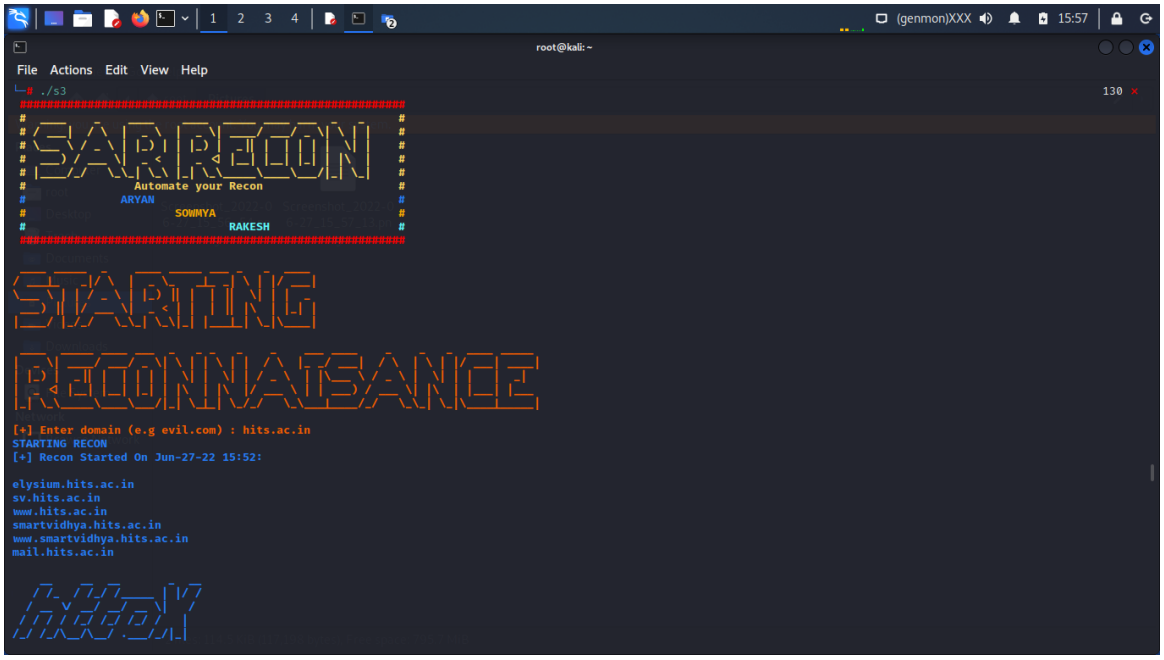


Figure 19: recon started


Figure 20: finding the services of live subdomains.


Figure 21: finding the services of live subdomains.


Figure 22: filtering working domains and failed domains.

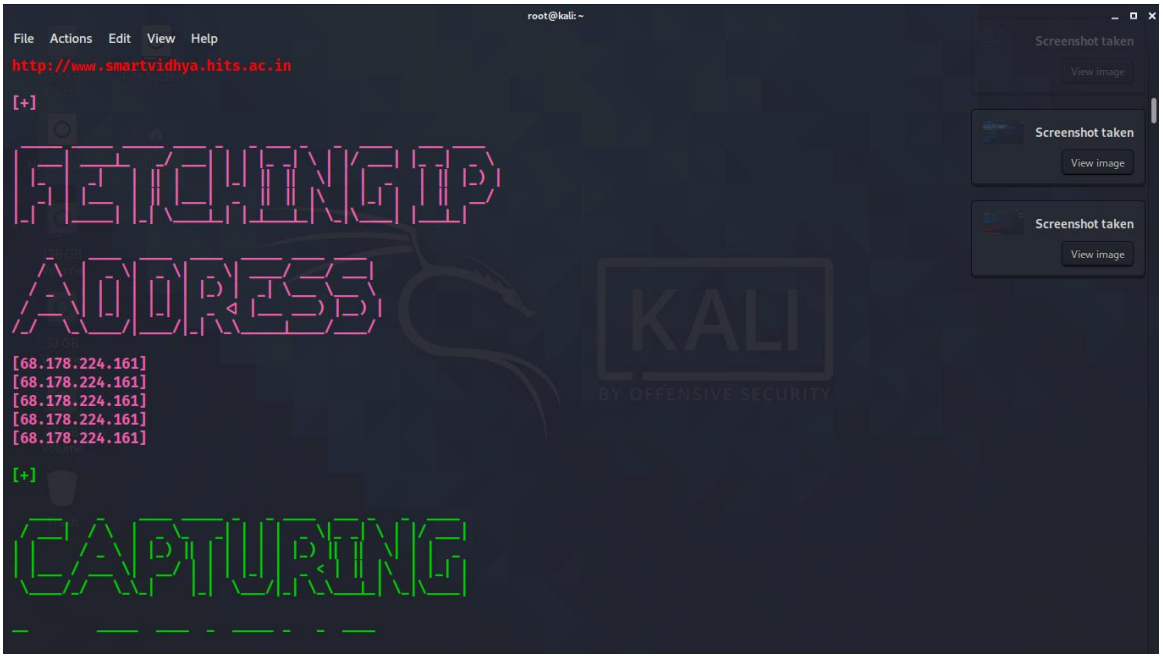


Figure 23: fetching ip address of working domains.


Figure 24: Screenshotting working domains.


Figure 25: screenshotting non-working domains.

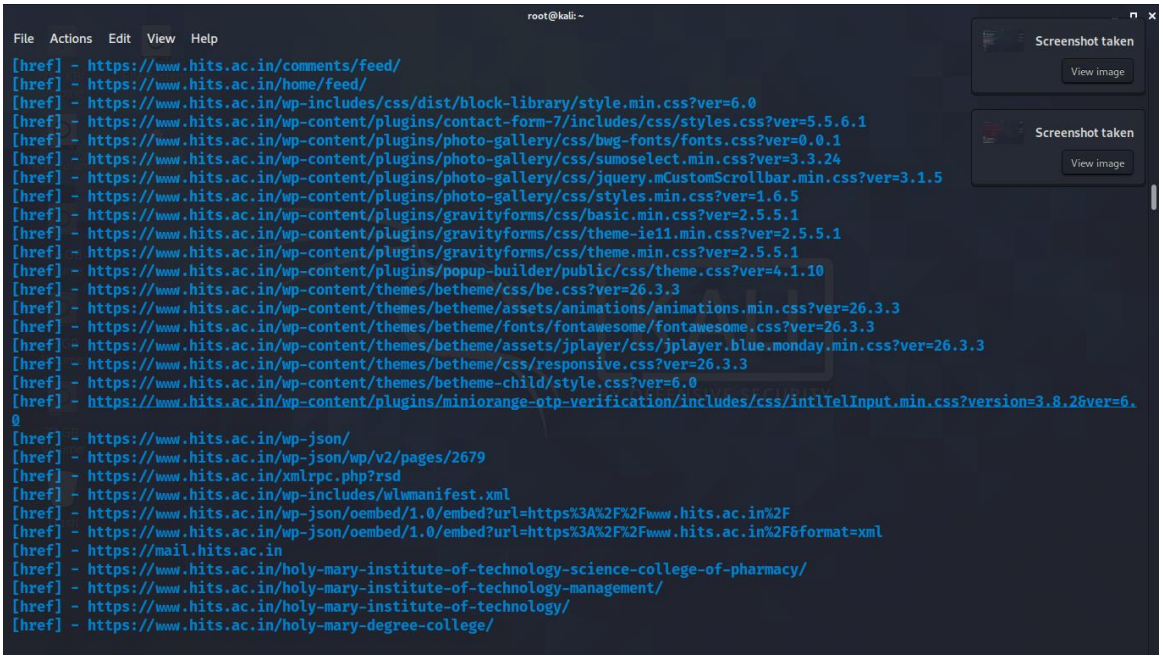


Figure 26: fetching all url branches of each and every working domain.

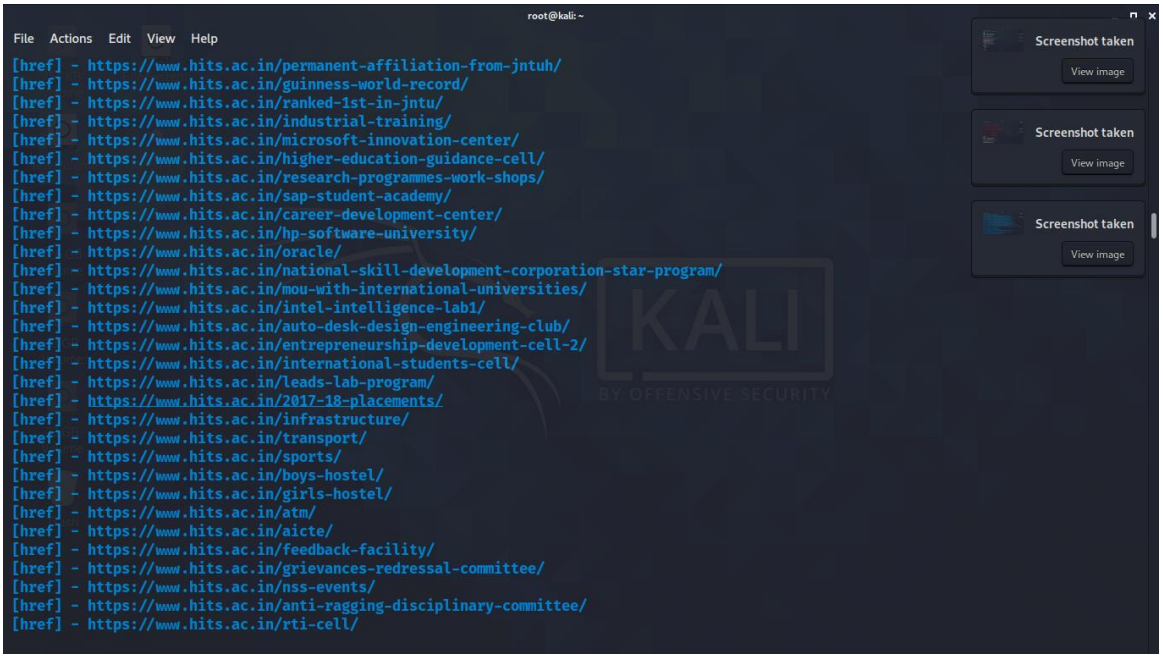


Figure 27: fetching url directories.

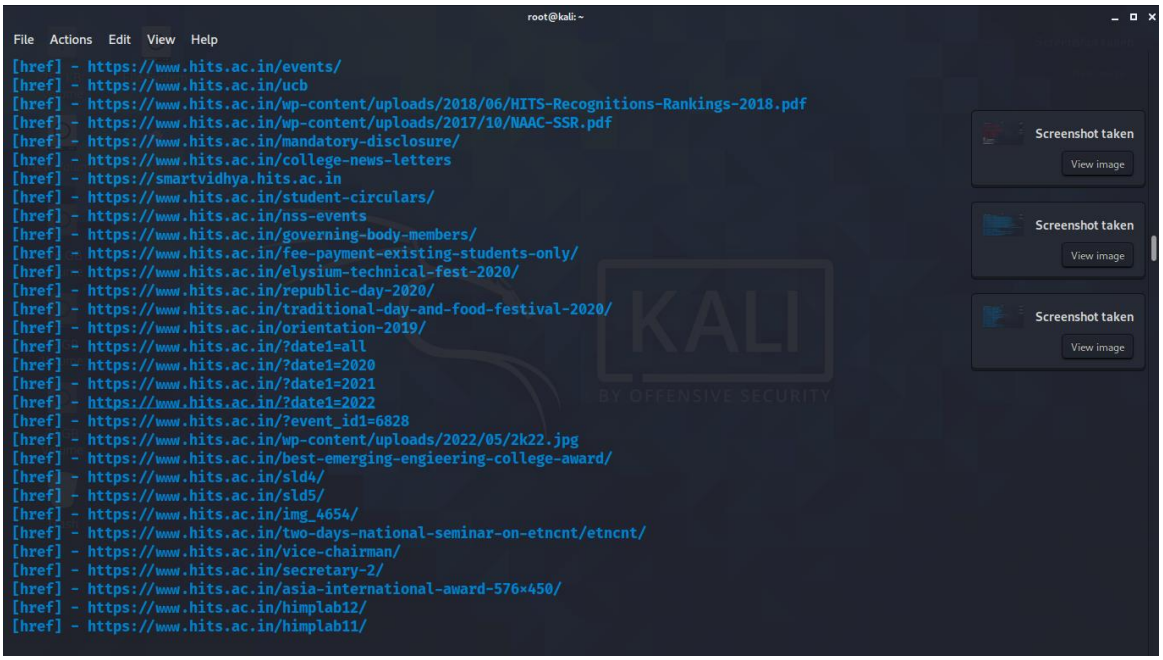


Figure 28: fetching url directories.

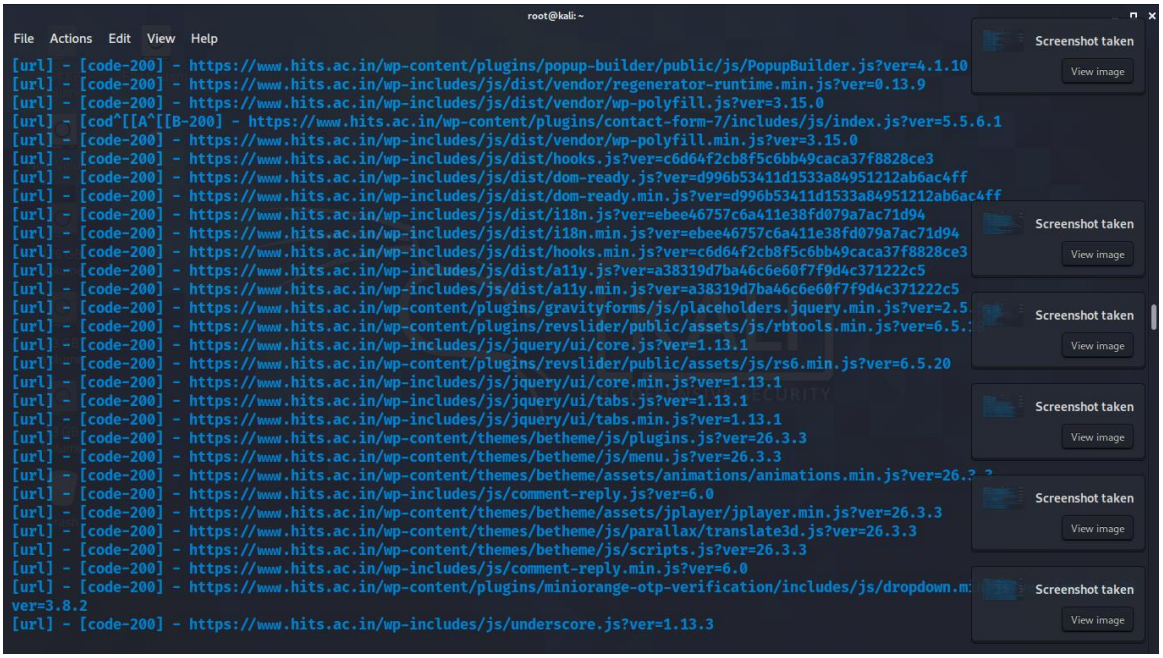


Figure 29: fetching url directories.

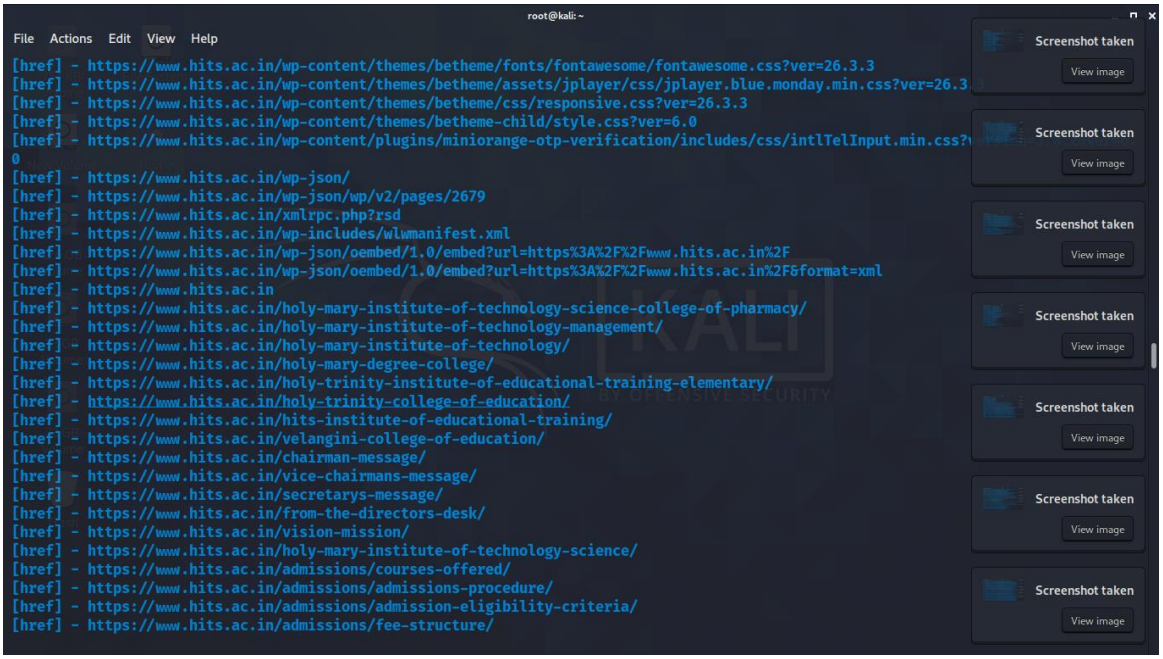


Figure 30: fetching url directories.

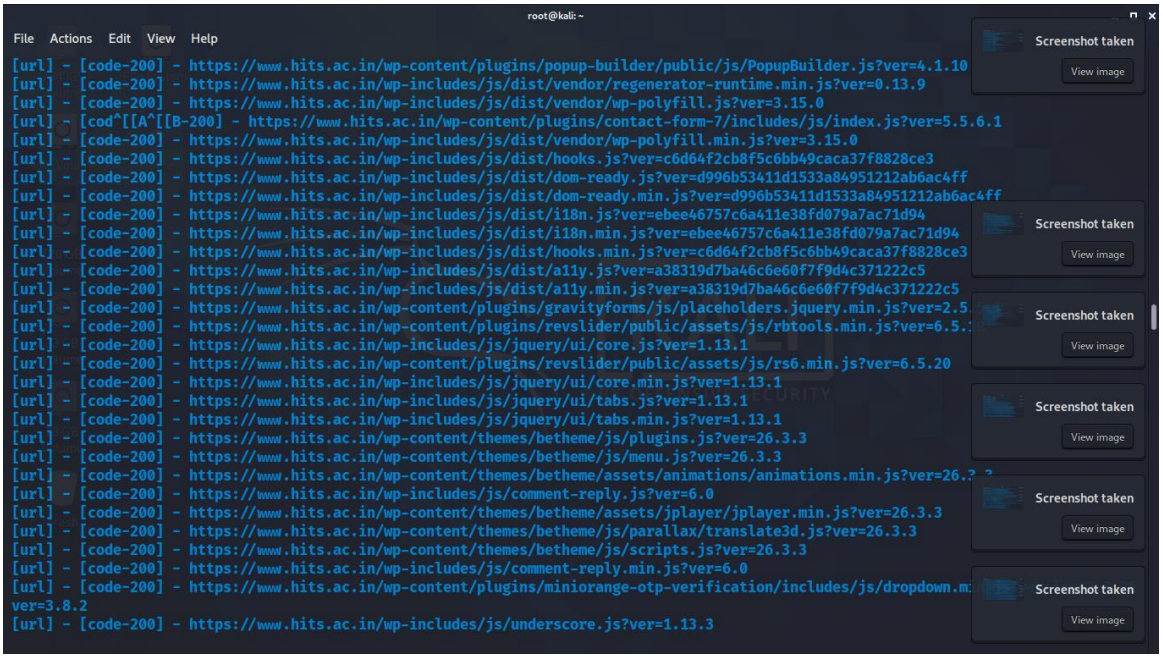


Figure 31: fetching url directories.

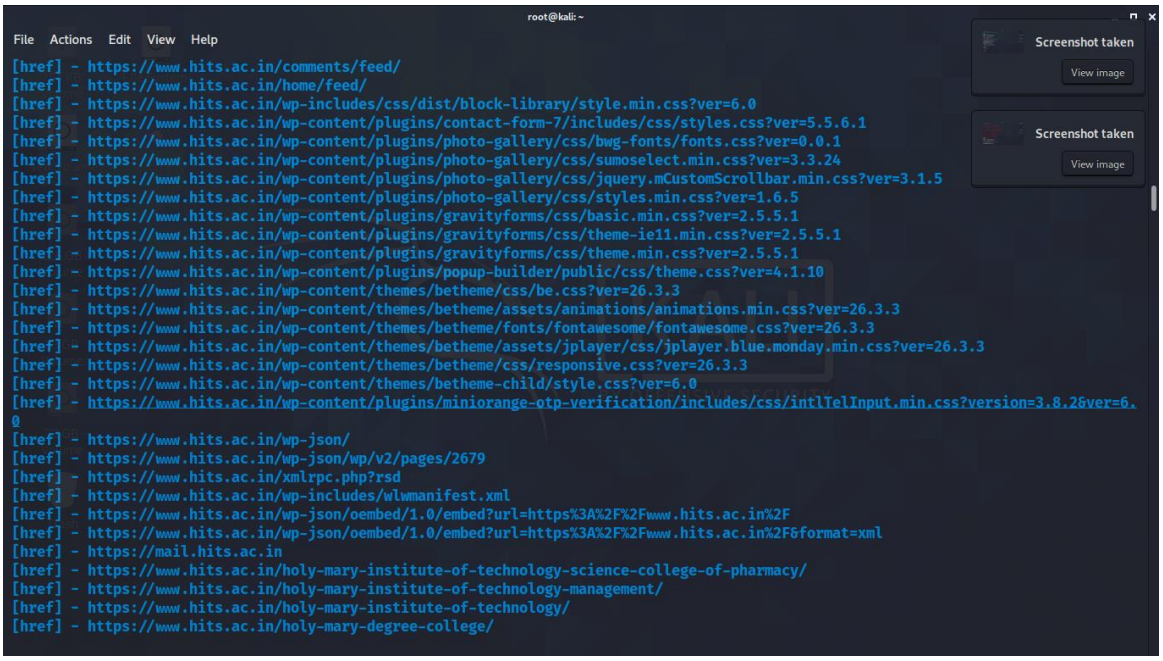


Figure 32: fetching url directories.

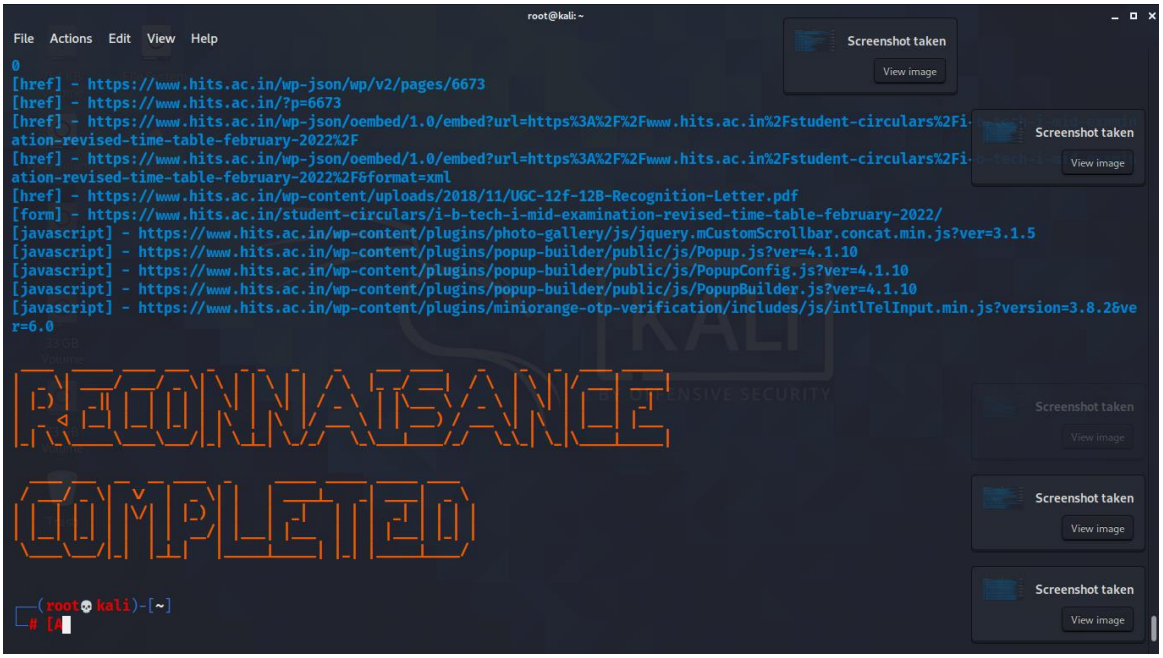


Figure 33: fetching url directories and reconnaissance completed screen.

# 6. SYSTEM TESTING

## 6.1 INTRODUCTION

The system provides secured and accurate record of data about gathering the information about an organization. The main pillar of reliability of the system is the backup of the database, which is continuously maintained and updated to reflect the most recent changes. The overall stability of the system depends on the stability of the application and the databases.

The system should be available to find the large scope data; means the user can access the previously stored data anytime and anywhere. In case of a hardware failure or database corruption, a replacement interface will be shown. Also, in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted.

The system is very easy to maintain, and all cached data will be rebuilt during every start-up. There is no recovery of user data if it is lost, Default values of system data will be assigned when necessary.

Port scanning is one of the most popular forms of reconnaissance ahead of a hack, helping attackers determine which ports are most susceptible. Port scanning can lead to a hacker entering your networ0k or stealing proprietary data.

Port scanning provides the following information to attackers:

What services are running.

- Which users own the services.
- If anonymous logins are allowed.
- What network services require authentication.

During a port scan, hackers send a message to each port, one at a time. The response they receive from each port determines whether it's being used and reveals potential weaknesses.

Port scans send requests to every port, asking to connect to a network. The scan then makes note of the ports that respond and which seem vulnerable.

Once the attacker has determined vulnerable ports in a network, the scan will classify ports into three categories:

- Open: The host responds, announcing it is listening and open to requests. An open port means it's a path to attack the network.
- Closed: The host responds, but notes there is no application listening. Often, hackers will come back to scan again in case it opens up.
- Filtered: The host does not respond to a request. This could mean the packet was dropped due to congestion or a firewall.

In order to defend your network against port scans, it's important to understand the different types of port scans that hackers use.

- Vanilla: The scanner tries to connect to all 65,535 ports.
- Strobe: A more focused scan, looking for known services to exploit.
- Fragmented Packets: The scanner sends packet fragments as a means to bypass packet filters in a firewall.
- User Datagram Protocol (UDP): The scanner looks for open UDP ports.
- Sweep: The scanner pings the same port across more than one machine to see which computers are active.
- FTP Bounce: The scanner goes through an FTP server to disguise the source.
- Stealth: The scanner blocks the scanned computer from recording the port scan.

# 7. CONCLUSION

Reconnaissance is significant aspect of any hacking activity. Any data that a programmer can find out about the target can help in recognizable proof of potential assault vectors and focusing on endeavors to possible weaknesses. By utilizing a blend of latent and dynamic observation devices and producers, a programmer can augment the data gathered while limiting the likelihood of discovery.

At the end of both stages of reconnaissance, attackers will have enough information to proceed or cancel a cyber-attack. From an external reconnaissance, they will know the behavior of users and use it to an organization's disadvantage. The aim is only to find some form of weakness that attackers can then use to gain entry to the networks or systems of an organization. Internal reconnaissance, on the other hand, will enable attackers to learn more about the network in question. Some of the discussed tools are extremely powerful and give so much information that it could be thought of as being leaked by the network designers themselves. The attackers become knowledgeable about the vulnerabilities they can exploit within a network or system of an organization. At the end of this stage, attackers are then able to engage an organization on two fronts: either from the users' side or internally from the network's vulnerabilities.

# 8. REFERENCE


**[1]** Nagendran K, Adithyan A, Chethana R, Camillus P, Bala Sri Varshini K B "Web Application Penetration Testing," at International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-10, August 2019.

**[2]** Ahana Roy, Louis Mejia, Paul Helling, Aspen Olmsted "Automation of Cyber Reconnaissance: A Java based open-source tool for information gathering", published at 2017 12th International Conference for Internet Technology and Secured Transactions (ICITST).

**[3]** KristianBeckers, Sebastian Pape, Peter Schaab, Daniel Schosser, "Conference: International Conference on Trust and Privacy in Digital Business", August 2017.

**[4]** Usman Dar, ArsalanIqbal, "The Silent Art of Reconnaissance: The Other Side of the Hill", January 2018.

**[5]** Awan, J., and S. Memon, "Threats of Cyber Security and Challenges for Pakistan". in Proceedings of 11th International Conference on Cyber Warfare & Security. 17-18 March 2016.

**[6]** Adebiyi, A., Arreymbi, J., and Imafidon, C. 2012. "Security Assessment of Software Design using Neural Network, International Journal of Advanced Research in Artificial Intelligence.

**[7]** Rehman, S., Mustafa, L. 2012. "Software Design Level Vulnerability Classification Model", International Journal of Computer Science and Security 2012.

**[8]** Giovanni, A., Cagalaban, Song, J., Jung, S., Kim, S. 2009. "Software Vulnerability Design and Approaches for Securing SCADA Control Systems," International Journal for smart home.

**[9]** Alshammari, B., Fidge, C., and Corney, D. 2016. "Developing Secure Systems: A Comparative Study of Existing Methodologies," Lecture Notes on Software Engineering.

**[10]** Ghosh, A., McGraw, G. 2003. "An Approach for Certifying Security in Software Components", Current Issues in Education.

**[11]** Khan, M., Zulkernine, M., 2009. "Survey on Requirements and Design Methods for

Secure Software Development", Queen's University, Technical Report No.

**[12]** Yoshioka, N., Washizaki, H. Maruyama, K. 2008. "A Survey on Security Patterns", Progress in Informatics.

**[13]** Jain, S., Ingle, M. 2011. "A Review of Security Metrics in Software Development Process", International Journal of Computer Science and Information Technologies.

**[14]** Xie, Y., Aiken A. "Static Detection of Security Vulnerabilities in Scripting Languages," in Proceedings of the 15th conference on USENIX Security Symposium, 2006.

**[15]** Turner, S. 2012. "Security vulnerabilities of the top ten programming languages: C, Java, C++, Objective-C, C#, PHP, Visual Basic, Python, Perl, and Ruby,"

**[16]** Javier Paster-Galindo, Pantaleone Nespoli, Felix Gomez Marmol, Gregorio Martinez Perez, "The Not Yet Exploited Goldmine of OSINT: Opportunities, Open Challenges and Future Trends", January 2020

**[17]** Barnum, S., Sethi, A. 2017. "Attack Patterns as a Knowledge Resource for Building Secure Software", Cigital, Inc.

**[18]** Bowman H. Miller, "Open-Source Intelligence (OSINT): An Oxymoron?", December 2018.

**[19]** Baig, M. 2012. "Security Vulnerabilities in PhP Applications", San Diego State University.

**[20]** Lewis, J., Timlin, K., "Cyber security and Cyber warfare," The Center for Strategic and International Studies (CSIS), Wahington DC. 2011.

**[21]** Sushmita Reddy Mamilla, "A Study of Penetration Testing Processes and Tools", May 2021.D.C. Ukpabi, 'Statistical Analysis on Crime Rate in Nigeria',2018.

**[22]** O. Faweya, A.T. Adeniran, and K.O. Balogun, 'Principal Component Analysis of CrimeRateinNigeria: ACaseStudyofEkitiandOsunState', Am.J.Math.Stat., vol.8, no.4, pp.79–88,2018.