HACKING WITH KALI LINUX

A Beginner's Guide to Ethical Hacking with Kali & Cybersecurity, Includes Linux Command Line, Penetration Testing, Security Systems and Tools for Computer

STEPHEN FLETCHER

HACKING WITH KALI LINUX

A Beginner's Guide to Ethical Hacking with Kali & Cybersecurity, Includes Linux Command Line, Penetration Testing, Security Systems and Tools for Computer

STEPHEN FLETCHER

Hacking with Kali Linux

A Beginner's Guide to Ethical Hacking with Kali & Cybersecurity, Includes Linux Command Line, Penetration Testing, Security Systems and Tools for Computer

Table of Contents

Introduction

Congratulations on downloading Hacking with Kali Linux: The Ultimate Beginners Guide for Learning Kali Linux to Understand Wireless Network & Penetration Testing. Including how to Getting Started with Scripting and Security, and thank you for doing so. The book covers the numerous tools in Kali Linux that you can use for performing penetration tests. You will also be able to learn the operations of the various utilities in this Debian distribution. To use this book effectively, you will require prior knowledge of basic Linux administration, computer networking, and the command utilities, albeit on a minimum. This will help you to comprehend the various subjects that have been covered herein.

You will get to know how what makes it possible for hackers to gain access to your systems and the methods they use to steal information. Furthermore, you will also learn the countermeasures required to safeguard yourself against the numerous hacking techniques. To this end, the book covers topics that include: Basics of Hacking, Cyber Attacks, Linux for Hacking, Basics of Kali, Scanning and Managing Networks, File and Directories Permissions, Cyber Security, Becoming Secure and Anonymous, and some basics cryptography that you will be required to know as an aspiring hacker.

Upon completion of this book, you will have become knowledgeable about both theoretical and practical concepts about basic hacking techniques. You will have the techniques needed for penetration of computer networks, computer applications alongside computer systems. Let me say that we have numerous books that cover this topic, but you have decided to pick this one up. Many thanks for that. No efforts have been left to ensure that the content in this book is relevant and refreshing to you. Have fun reading!

Chapter 1: Basics of Hacking

Definition of Hacking

This is a process of identification of flaws that are present in a given network or computer systems that can be used to exploit its weaknesses to gain access.

An excellent hacking example is employing the use of a password cracking algorithm to secure entry into a system. In this age, computers are indispensable when it comes to running successful businesses. Additionally, computers need to be networked to be able to facilitate the exchange of communication with other external businesses. This means that isolated computer systems on their own are not enough. By networking them, it means that we are exposing them to the outside world and thus making it possible for them to get hacked. Hacking essentially implies the use of computers to carry out malicious acts, for instance, stealing personal or corporate data, privacy invasion, fraud, and so on. Cybercrimes are known to cost organizations all around the world millions of dollars each year.

It is therefore prudent that businesses protect themselves against such attacks. Most of the hacking worldwide is carried out with criminal intent. This can range from committing some form of fraud to ruining the reputation of the targeted organization. Hackers can steal crucial data, embezzle funds, and even spread misleading or malicious information that

can be detrimental socially. Hacking is a crime and is in most jurisdictions, punishable by law. In spite of this, there is a form of hacking that is considered beneficial. This is done by professionals, government law agencies, and other accredited institutions. Primarily, they intend to counter the malevolent intent of malicious hackers. This way, it is possible to safeguard systems against harm. The protection and safety of the general society and its citizens can be achieved by this type of professional hacking, otherwise known as ethical hacking.

Common Hacker Attacks

The following are the most common types of hacker attacks against computers and networks.

1. Denial of Service (DoS) Attack

A websites' server can get overloaded when it is flooded by traffic more than it can handle. Picture this, a road designed to handle traffic from a small town can quickly get gridlocked when there is an influx of external traffic. The users will experience massive delays, and the inconvenience will be great. This is how a denial of service attack affect websites. The additional traffic on the site will make it impossible to provide service to visitors who are trying to access it.

A practical example is a newspaper's website carrying breaking news. Many people will try to access it to find out more, consequently overloading the site. In a DoS attack, however, the overloaded traffic is ordinarily malicious. The intention is to shut down the website from its legitimate users. A Distributed Denial-of-Service Attack (DDoS) is an attack carried out by many computers at the same time. It is challenging to cope with this type of attack since the IP addresses will appear to be originating from many different locations around the world simultaneously. This means that it is difficult to determine the source of attack by network administrators.

2. Cross-Site Scripting (XSS)

An attacker can go after a vulnerable website in an SQL injection attack. Stored data can be targeted. For instance, sensitive financial data, user credentials, among others. A cross-site scripting attack is preferable to an

attacker who would instead directly target a website's users. Just like an SQL injection attack, a cross-site scripting attack involves injecting malicious code into a site. The only difference is that the website itself is not being attacked. What happens is that a malicious attacker will carry out an injection on the user's browser upon visiting the infected site. A common way to do this is by injecting the code, which is malicious into a comment or a script that could automatically run. For instance, in JavaScript, a link can be embedded in a comment on a blog. This type of attack can, in essence, damage a website's reputation by risking users' information without necessarily doing anything malicious. In some cases, sensitive information users transmit on the site can be hijacked through cross-site scripting before even the owners of the website realize that there is a problem.

3. SQL Injection Attack

SQL is a programming language we use to communicate with databases. Most servers doing storage of critical data for websites and services usually utilize SQL for the management of data that is stored on their databases. This type of assault targets, precisely this kind of server. It employs the use of a malicious code to prompt the server to disclose the information it would not normally do. The problem can be amplified if the server is used for storing private customer details, for example, usernames, numbers of credit cards, and so on. This information can be used to identify a person. The attack carries out its intended goal through the exploitation of any one of the known SQL vulnerabilities that are known to permit running malicious codes on the SQL server. For instance, an injection - attack - vulnerable SQL server, may motivate an attacker to type in a code in the website's search box, which will make the SQL server of the site to dump the usernames and passwords that have been stored for the site.

4. Phishing

A good number of phishing scams come in the form of text message and email campaigns that are meant to create a feeling of curiosity, compelling urgency, or even instill fear in potential casualties. The victims are then prodded into disclosing information that is deemed sensitive, following

the links to malicious websites by clicking on them, or by downloading and opening unknown attachments containing malware. For instance, users of a particular online service can receive an email that alerts them of a policy violation that requires an action to be done immediately on their part. An example of such a move is a password change. Upon clicking the link, the user will be redirected to an illegitimate website that is almost identical in every aspect as the legitimate one, prompting an unsuspecting user to input his or her credentials to the site. That information is sent to the attacker once the form is submitted.

5. Malware

Attackers usually prefer to deploy malware in a users' computer so that they are in a position to gain a foothold there. It is one of the most effective ways of gaining access. First, let us define what malware is. It can be said to represent various configurations of software that are harmful, for instance: ransomware and viruses. A malware present in your computer is capable of wreaking all sorts of havoc. This includes but is not limited to the following: monitoring and recording actions and keystrokes

you perform on your computer, assuming control of your machine, and even sending your confidential data to the attacker's base directly from your computer. There are many ways through which an attacker can deliver malware into your computer. However, for this to work, it will require that the user, on their own volition, take action to install the malware such as opening an attachment looking harmless or clicking a link to download a particular file all of which contain hidden malware in them.

6. Session Hijacking and Man-in-the-Middle Attacks

The computer you are using typically make numerous back-and-forth transactions with the servers around the world to inform them of your identity and the specific websites or services that you are requesting whenever you are on the internet. If all goes well, you will get all the information that you had requested from the web servers. This is the norm both when you are logging into a particular website with your credentials, that is, the username and password or just simply browsing. A unique session ID is given to the session between the remote web server and your computer. That ID needs to stay private. When an attacker comes into the picture, they can use the obtained session ID's to hijack the session.

That is made possible through capturing the session ID and feigning resemblance to the computer making a request. The attacker can log in the same way as an unsuspecting user would do and, as such, obtain access to crucial information on the webserver. We have several ways an attacker can employ to be able to steal the session ID. One of them is a cross-site scripting attack that we have discussed before. Alternatively, the attacker may decide to do session hijacking by placing themselves in between the remote server and the requesting computer. Here, they will be pretending to be the other party involved in the session. This way, they will be able to intercept information being transmitted from both directions. This is what we refer to as a man-in-the-middle attack.

7. Password attack

The commonly used method of authentication of users to any information system is the use of passwords. This means that stealing passwords is an effective approach that can be used in the attack. To do this, all it takes is

to look around a targets' desk then carry out a sniffing operation on the connection. This will enable you to obtain unencrypted passwords. Here, one can use social engineering tactics to gain access to a database containing passwords or do outright guesswork to get the password. This can be carried out in either a random or systematic fashion. We have the following types of password attacks, brute-force attacks, and dictionary-based attacks. In the former type of attack, different passwords are tried randomly with the hope one of the combinations will work. An attacker can apply some logic to this procedure, such as trying passwords that relate to a users' name, hobbies, job title, and so on.

On the other hand, a dictionary-based attack utilizes common passwords found in a dictionary in their attempt to access a target's network or computer. A commonly used technique is whereby an encrypted file containing passwords is copied, and then similar encryption is applied to the password dictionary. The passwords are then compared. To safeguard oneself against a dictionary or a brute-force attack, all that is required is to put in place an account lockout policy. This will mean that a particular account will be locked after exceeding a specified number of attempted logins.

8. Eavesdropping attack

This attack is carried out by intercepting the traffic on the network. Through the use of eavesdropping, a malicious user will be able to get access to credit card numbers, passwords alongside other information that is deemed to be confidential in which the user may be exchanged over the network. We can either have a passive or an active eavesdropping attack. Let us briefly look at what they are.

Passive eavesdropping — In this mode of eavesdropping, a malicious user will attempt to obtain information by way of listening in on the messages being transmitted over the network.

Active eavesdropping — Information being transmitted is actively captured by way of an attacker disguising as if they were a friendly unit. This is carried out through querying the data transmitters. We call this scanning, tampering, or simply probing.

It is prudent to note that the detection of passive eavesdropping attacks is crucial. This is because it is a precursor to the active attacks. Data encryption is the best countermeasure for eavesdropping.

What Are the Types of Hackers?

Let us differentiate the different types of hackers we are likely to encounter below. You will notice that each category of hackers possesses different objectives. We will also look at the various roles and goals each of the hackers has.

a) Black Hat Hacker

Commonly known as black hats, they usually have extensive knowledge regarding the various methods about bypassing security protocols and breaking into computer networks. Malware is generally written by black hats to help gain access to these systems. A black hats' main goal is usually to make a personal or financial gain. Some of them do carry out cyber espionage while some do it for fun. This category of hackers ranges from inexperienced amateurs whose idea of fun is spreading malware, to those that are experienced whose objectives are to embezzle privileged data. Many a time, it is financial information they seek. They are also interested in harvesting login credentials and personal information.

Besides stealing data, they usually alter or sometimes destroy the data they have obtained if it does not serve their purpose.

b) Grey Hat Hacker

These can be said to be neither the bad or good guys! Grey hats are neither white nor black. This category of hackers has the characteristics of both black hats and white hats. Most of the time, grey hats will attempt to unearth, without the permission or knowledge of the owners, vulnerabilities that are present in a system. If they discover an issue, they will report the same to the owners. Most of the time, they will ask for some financial compensation so that they can fix the problem. In the 'unfortunate' event that the owner does not comply, they will go ahead and post their exploits on the internet for everyone to see. Grey hats' intentions are not necessarily malicious; all they want is to make some dollars out of their discoveries. After finding vulnerabilities, Grey hats will not usually exploit the vulnerabilities that they have unearthed. What

makes this grey hacking illegal is the fact that no prior permission was sought from the owners of a particular system the hackers targeted. For the readers seeking to become hackers, it is essential to note that not all hackers are created equal. We have white hat hackers who are always trying to uncover and fix vulnerabilities before black hats find them. This way, we have a lot fewer cyber-crimes now.

c) White Hat Hacker

Unlike the two previous types of hackers, white hat hackers are those that put their skills to good use. Their intentions are benevolent. These are the good guys. This group of hackers is commonly referred to as ethical hackers. They may be an organizations' employees or work as contractors in information security companies. They usually try to discover vulnerabilities in a system through hacking. The methods used in the hacking process are similar to those that black hats use, but there is one differentiating aspect. White hats first seek the owners' consent. This makes the hacking legal. White hat hackers usually carry out penetration tests; they do assess vulnerabilities and conduct in-place testing of a company's' security systems. We even have certifications, training, courses, and conferences that are hosted ethical hacking.

What Is Ethical Hacking?

We define ethical hacking as an approved implementation whereby there is bypassing of a systems' security that helps in the identification of threats and potential data breaches in a network. In this scenario, an organization that owns a particular network or system grants permission to cybersecurity experts to carry out such activities to test the defenses a system has put in place. This, therefore, implies that, unlike malicious hacking, ethical hacking is a legal process that has been planned and approved. The main goal of an ethical hacker is to scrutinize a particular network or system for weak points. It is via these weak points that malicious hackers use in their exploitation or destruction.

While at it, they do gather and perform an analysis of the information. This will help them in the planning process of the organization's IT infrastructure. In doing so, the security footprint will be significantly improved in a manner that it can withstand or divert attacks. The demand for ethical hacking has witnessed a dramatic increase in recent times due to the growth of the information security sector. Ethical hacking is also known as white hat hacking. Ethical hacking is the practice of attempting to infiltrate and exploit a system to find out its weaknesses so that it can be better secured. We can categorize it into two broad classes. These are, penetration testing; this is mostly for a legitimate information security

firm and; hacking by intelligence agencies or a nation's military. There is a rising demand for hacking in both areas.

- **Penetration Testing**

This is a mechanism that is utilized by organizations to ascertain the robustness of their security infrastructure. Here, security professionals will play the role of the attackers, whereby they will attempt to discover flaws and vulnerabilities in a system before the malicious fellows do. One key objective is the identification and reporting of vulnerabilities to companies and organizations. As organizations become increasingly security conscious and the cost of security breaches rises exponentially, many large organizations are beginning to contract out security services. One of these critical security services is penetration testing. A penetration test is essentially a legal, commissioned hack to demonstrate the vulnerability of a firm's network and systems. Generally, organizations conduct a vulnerability assessment first to find potential weaknesses in their network, operating systems, and services. I emphasize potential, as this vulnerability scan includes a significant number of false positives (things identified as vulnerabilities that are, in reality, not vulnerabilities). It is the role of the penetration tester to attempt to hack, or penetrate, these vulnerabilities. Only then can the organization know whether the weakness is real and decide to invest time and money to close the vulnerability.

- **Espionage and Military**

Cyber espionage can be said to be the practice of accessing information and secrets without the knowledge and permission of the entities being targeted. They can be ordinary individuals, rivals, competitors, groups, governments, or even enemies. The objectives here are broad. They can be political, economic, personal, or even military-related. The techniques used, too, are diverse. Hackers can use malicious software, cracking techniques, proxy servers, among others, to attain their stated objectives. Espionage can be carried out online by professionals from their computer desks, or it can be done by infiltration using trained moles and conventional spies. In some circumstances, it can be carried by amateurish hackers with malicious intent and software programmers. It is common knowledge that every nation on earth carries out some form of cyber

espionage or even cyber warfare, albeit covertly. Gathering intelligence on military activities of other countries has been made more cost-effective by hacking. Thus, a hacker has their place cut out in the defense systems of any nation.

Chapter 2: Cyber Attacks

Definition of a Cyber Attack

A cyberattack can be said to be the intentional exploitation of computer systems, enterprises that depended on networks and technology. Cyber-attacks employ the use of codes that are malicious for purposes of modification of computer data, code, or logic. All these have a net effect of disrupting repercussions, which often lead to compromise of crucial data. Cyber-attacks additionally are a platform for launching or committing cybercrimes. For instance, theft of information and identity. The other name of a cyberattack is a computer network attack. Cyberattacks may have the following ramifications:

- Unauthorized access and theft of intellectual property
- Web browser exploitation
- Infiltration of systems
- Breaching controls of access
- Stolen hardware,
- Fraud, identity theft, extortion
- DoS and DDoS attacks
- Sniffing of passwords
- Defacement of websites

Why Cyber-Attacks Are Crucial

We have recently seen how the consequences of cyber-attacks can be devastating. Running costs of businesses are increasing due to cyber-attacks. Data breaches are also costly to handle. A study conducted by the Ponemon Institute in 2018 found the average costs of a data breach to be 3.86 million dollars. However, it is not just about the financial aspects of cyberattacks. They can also: -

- Destroy reputations of the brands involved
- Erosion and in some cases decimation of the loyalty customers had for the brand
- Cyber-attacks can result in the loss of intellectual property
- Companies, in severe cases, can be run out of business
- They can also bring about penalties from the various regulatory agencies
- The security of states and governments can be significantly impaired.
- There is a chance that future attacks will likely be carried out.

Malware Attack

A malicious piece of software is unwanted software that, without your consent, is installed in your system. Such software is known to attach themselves to a legitimate software or code, giving it a platform to propagate itself. Malware can persist in applications we use on a day to day basis or can even replicate itself over the Internet. Let us discuss the common types of malware in existence today:

1) **Macro viruses** — They are known to infect commonly used programs. For instance, Microsoft Excel, Microsoft Word, and so on. These types of infections bind themselves to the initialization sequence of the applications. Upon the opening of an application, the macro virus will affect instructions before transferring control to the application. Just like many other viruses, they are capable of replication through attachment to other codes or programs running in the computer system.

2) **File infectors** — These ones typically bind themselves onto an executable code. Once the code is loaded, the virus will be installed. A different type of file infector many a time associates itself with a file on the computer through the creation of a virus file that possesses a similar name but is appended with a .exe extension. Upon opening the file, the virus code will execute.

3) **System record/boot-record infectors** — they work just like the file infectors only that they attach themselves to the master boot record that is found on hard disks. They will load viruses into the memory of the system once the system is started. From there, the infectors will propagate to other computers or other disks.

4) **Polymorphic viruses** — These normally operating by way of concealing themselves by using varying cycles of encryption and decryption. The virus that is encrypted together with its associated mutation engine is, at the onset, decrypted using a decryption program. After decryption, it will proceed to infect a small portion of code. The mutation engine will be used to build a new decryption routine that the virus will encrypt. It is this package of mutation engine and virus that is encrypted that will attach itself to a new code. The process is repetitive. It is quite challenging to detect such viruses. They possess a high level of entropy due to the numerous alterations to their source code. Utilities like the Process Hacker or most anti-virus software can detect and isolate them using this feature.

5) **Stealth viruses** — These are viruses that capture the functions in a system so that they can hide. This is made possible by the stealth viruses interfering with software that detect malware. The result here is that the software will wrongly find an infected area as being uninfected. Stealth viruses hide any size increases of a file that is infected or sometimes even alter the infected file's time and date of last modification.

6) **Trojans** — Famously known as Trojan horses, these are programs that conceal themselves within a commonly used but different program. A trojan horse, in most cases, carries with it has a

malicious function. Trojans, unlike viruses, do not self-replicate. Besides launching attacks on target systems, Trojans can create back doors that may be used by attackers to exploit a system. A Trojan can be programmed, for instance, to open a port that is high-numbered through which a hacker can listen and even carry out an attack in the future.

7) **Logic bombs** — This type of malware usually is bound to an application. For it to work, a specific occurrence has to trigger it. That may be a predetermined logical condition, a set time, a specified date, and so on.

8) **Worms** — This is a program that is self-contained and can propagate across computers and networks. This implies that they do not attach themselves to a host file viruses. They are frequently spread through attachments that are sent on emails. A user who downloads and opens the attachment unknowingly activates the worms. One exploit of worm comprises of the worm replicating to every contact that is present on the email address of the infected computers. They can also spread across the internet resulting in the overloading of email servers. This scenario leads to a denial-of-service to the various nodes that are present on the network.

9) **Droppers** — These are programs that can be used to install viruses on victims' computers. Many a time, the dropper itself does not usually have malicious code. This implies that they are therefore not detectable by software responsible for virus-scanning. A dropper, however, can connect to the internet where it can download updates to an existing virus software resident on the system that has been compromised.

10) **Ransomware** — This form of malware that works by preventing access to the targets' data. Sometimes, ransomware threatens typically to delete the data or publish unless the owner of the ransomware is paid. We have those that are easy to reverse for a knowledgeable person and those that are more advanced. These use a cryptoviral extortion technique. The technique does encryption of

the victim's files. That makes it impossible to recover them unless the victim obtains a decryption key.

11) **Adware** — companies use this is a software application for purposes of marketing. Whenever any program is running, you will see advertising banners being displayed. Adware can be downloaded to your system automatically when you browse the website, which can be viewed through pop-up windows or via bars that automatically come up on the computer screen.

12) **Spyware** — A spyware is a program that can be installed on a system primarily to gather information about computers, users, and even their browsing mannerisms. Spyware can track almost everything you do, both offline and online, without your knowledge. The collected data is then transmitted to a remote user. Just like droppers, spyware can connect to the internet where it can download other malicious programs and install them on the victims' computer systems. This program works in the same way as adware with an exception that it usually comes as a separate stand-alone program. It is installed when you are doing an install of a freeware application, and in most cases, it happens without your knowledge.

MAC Spoofing

We can define it as an unsanctioned alteration of a devices' MAC address. Simply stated, it is the falsification of a MAC address of a network device that is within a given computer network. This way, a malicious user can use fake identification, that is, the MAC address, to pass off as if it were one of your own devices and, subsequently, carry out an interception of communication being exchanged in the network. The falsification of a devices' MAC address can be done in the following ways:

- A simple MAC address change
- Creating a MAC address in a random fashion
- Using different manufacturers' MAC address
- Configuring a new MAC address while keeping the current manufacturer intact and then activating it automatically.

- **Reasons for Carrying out MAC Spoofing**

MAC spoofing can be carried out both for legitimate and non-legitimate reasons. The latter can be taking over another computer's identity, and the former can be used in the creation of wireless connections to a network. A different example of the legitimate use of MAC spoofing is the modification of the function of a single computer from a router to the computer and back to router through sharing a single MAC address.

- **Non-Legitimate Uses of MAC Spoofing**

Another example of an illegitimate use is when an intruder modifies their MAC address to enable them to gain access to a target network as if they were an authorized user. An attacker can wreak devastating damage to a network or system using this newfound identity. For instance, they can launch a denial of service attack on the computer systems, or in some cases, bypass the control mechanisms for access to paving the way for a more advanced intrusion. Also, an attacker can decide to alter their MAC address so that they can evade network intrusion detection systems. This makes them invisible to laid down security measures, thereby giving them plenty of time to act without being detected.

- **Lawful Uses of MAC Spoofing**

One legal use of MAC spoofing is whereby the function of a single computer is changed from being a router to a computer. The reverse also applies. That is, altering the function of a computer back to being a router. Suppose we have a single public IP address, we can only use it on one router or computer. Now, if we have two wide area network IP addresses, it implies that the MAC address of the two devices (two computers) have to be different. The computers may need to be regularly swapped for some reason to connect them to the cable modem. The exercise would be quicker and easier when the MAC addresses of the devices are changed instead of changing the Network Interface Card. To this end, numerous cable modem routers possess a "Clone MAC Address" feature that is inbuilt purposely for this.

A MAC address that is falsified enables attackers to bypass security mechanisms that have been set. For instance, an attacker will be able to impersonate real devices or even conceal themselves behind other devices on the network. To conclude, it is instructive to note that the exercise of MAC address changing is a legitimate technique that can be used to ensure a proper network operation besides being a fraudulent technique.

- **Protection Against MAC spoofing**

Network monitoring, analysis, and security are required to combat MAC spoofing. Management of access to a company's' network for purposes of

keeping out the bad guys is certainly a good idea. Guests should have restricted access to an organizations' network connection, say Wi-Fi. The main reason for this is that a significant portion of the MAC spoofing attacks originate or, instead, are carried out from an internal network. It is also proactive for a company to ensure no unauthorized persons are left behind at its premises and that visitors are never left alone. This measure will go along the way in preventing unauthorized people from manipulating or connecting to an internal network by using connecting directly to the ethernet utilizing a cable. It is advisable that companies also adopt IPsec technologies alongside communication encryption within the network system that will help to eradicate eavesdropping of existing MAC addresses. In more prominent companies, the use of active networking hardware that is advanced ensures that they have improved protection utilizing firewall configuration or switch configuration. These techniques block external incoming packets since they are likely an attack vector.

- **ARP Spoofing**

This is an attack whereby an entity deemed to be malicious transmits falsified ARP messages across a local network. The consequence of this is that the MAC address and IP address of a legitimate server or computer that is within the network will be linked. That means that an attacker will be able to receive any information and data that is intended for the authentic IP address. Additionally, an attacker is also able to modify, intercept, and sometimes stop the data being transmitted. This type of attack, however, is only possible on local area networks using something we call Address Resolution Protocol.

Examples of ARP Spoofing Attacks

ARP spoofing attacks have severe consequences for targeted enterprises. The main goal of these spoofing attacks is to steal information. This information is usually sensitive. They are not only limited to this but are also able to enable other attacks, for example:

- **DoS attacks** : Such attacks take advantage of ARP spoofing to connect many IP addresses to a single MAC address belonging to the target. Consequently, the traffic initially intended for the

multitudinous IP addresses will be rerouted to the MAC address of the target, thereby overloading it.

- **Session hijacking** : This attack uses ARP spoofing mainly for obtaining a session ID. The IDs can then be used to gain access to private data and even systems.
- **Man-in-the-middle attacks** : Just like session hijacking, MITM attacks normally intercept and alter the traffic between unsuspecting victims.

- **Detection, Prevention, and Protection Against ARP Spoofing**

One can detect and keep ARP Spoofing attacks at bay through one or more of the following:

- Making use of ARP spoofing detection software: Such programs carry out an inspection and certification of data before transmission. Data that seems to be spoofed is customarily blocked.
- Packet filtering: This is a useful technique that can be used to prevent ARP spoofing. Packets having conflicting source address information are filtered out and blocked.
- Encrypting network protocols: Protocols such as HTTP Secure (HTTPS), Secure Shell (SSH), Transport Layer Security (TLS), among others help to prevent attacks by ARP spoofing. They accomplish this by carrying out the encryption of data before it is transmitted and authentication of the same data upon receipt.
- Keep trust relationships to a minimum: Companies and institutions need to come up with security protocols that do not depend on trust relationships. In such types of relationships, the authentication mechanisms depend on IP addresses only. This makes it a lot easier for ARP spoofing attacks to occur.

Rogue DHCP Server

- **A Definition of DHCP**

We define it as a protocol the network uses to enables a particular server to be able to assign an IP address to a specific computer automatically from a range of numbers that are defined for a network.

- **So, What Is a Rogue DHCP Server?**

We define a rogue DHCP server as a DHCP server in which an attacker or an unaware user sets up on the network. The rogue DHCP server will not be under the network administrator's control. An example of an accidental rogue device is a DHCP capable modem that an unknowing user has connected to the network. The user, in most cases, is usually unaware of the consequences of doing so. Additionally, these servers are widely utilized by attackers for network attacks, for example; sniffing, reconnaissance and man in the middle attacks. Upon connection to the network, the two DHCP servers (legal and rogue) will provide the devices with everything they need to carry out communications. These can include Default gateways, DNS servers, WINS servers, and IP addresses, to name a few.

Occasionally, clients using IP addresses or gateways that are incorrect are bound to experience network access problems. Also, when a rogue DHCP functions as the default gateway to a machine that is controlled by a malicious or misbehaving user, it will be able to sniff traffic that is transmitted to other networks, thereby breaching user privacy and the network security policies in place. A virtual machine software is also able to function as a rogue DHCP server, albeit inadvertently. This happens when it is run on a client machine, which is linked to a network. Here, it will be issuing out IP addresses that are random to all the clients next to it on the network. The consequence is that large sections are disconnected from the Internet together with the remainder of the domain.

Prevention of Rogue DHCP Servers

Intrusion detection systems having the relevant signatures can be used to stop Rogue DHCP servers. Multilayer switches may also be set up in a way that they can drop the packets. DHCP snooping is perhaps the most common way of dealing with these servers. This works by dropping DHCP messages originating from an untrusted DHCP server

- **TLS/SSL Encryption**

The encryption is used for securing HTTP network connections. It safeguards the connections against interception and man-in-the-middle attacks that can be propagated through the web. TSL/SSL encryption increases the difficulty of the communications interception process happening between a server and the client.

- **Wi-Fi Encryption**

An open Wi-Fi network is more likely to be plagued by eavesdropping attacks. Wi-Fi encryption, therefore, provides the best way of keeping hackers at bay. This will mean that they will not be able to access any information that passes through the network. That being said, Wi-Fi encryption also has its weaknesses, and you should not place all your faith in that.

DDoS Attack

This is a malicious attempt that is solely intended to disrupt regular traffic of a server, network, or service that is being targeted. This is, in most cases, done by overwhelming a target alongside its surrounding infrastructure with a deluge of traffic from the internet. A DDoS attack is effective in the sense that it uses multiple compromised computer systems as its source of traffic for an attack.

- **How the Attack Works**

This type of attack requires gaining network control for machines that are online by an attacker for them to be able to implement an attack successfully. Malware is used to infect the devices and computers hence turning them into bots. That is, they become zombie-like. A group of bots is known as a botnet. The attacker will be able to control these botnets remotely. The botnets will remotely receive updated instructions that the attacker will send to them. Once the attacker identifies a target along with its IP address, the botnets will send requests to it. This will overload the goal resulting in the denial-of-service to normal legitimate traffic. Since each bot is a valid internet device, it becomes quite challenging to separate the regular traffic from the attack traffic.

Chapter 3: Linux for Hacking

A Brief Introduction

As you begin your journey towards being a hacker, you will realize that most of the professional and expert hackers make use of Linux/Unix in their trade. That aside, we have some types of hacks where Mac OS and Windows can be used. Software such as Zenmap, Metasploit, Havij, Cain, and Abel and others have Windows versions. Usually, when the applications are developed in Linux and later ported to Windows, they always lose some of their capabilities. That is to say; we have capabilities that are found in Linux but are not present in Windows. This is the reason that many hacker tools are designed and build for Linux. It is, therefore, essential that anyone who intends to be a professional hacker learns or has some basics of a Linux distribution like Kali.

Why Hackers Use Linux

Below are some of the main reasons why hackers prefer Linux.

1. Linux Is Open Source

Unlike the Windows operating system software, Linux distribution is an open-source. A Linux user has access to the operating systems' source code. This means that one can manipulate and change a Linux operating system to suit their needs at will. Supposing the user wants to make a particular system operate in ways besides those it was originally intended to, the ability to manipulate and change the source code is of paramount importance.

2. Linux Is Transparent

A thorough understanding of a users' operating system is required for one to carry out hacking effectively. It is also necessary to have a knowledge of the operating system one intends to hack. Unlike windows, we can see and even manipulate all the working parts of Linux. That is to say, Linux is entirely transparent. It is not easy to understand the inner workings of a windows operating system. The transparency aspect means working with Linux is more effective.

3. Linux Offers Granular Control

A Linux user has infinite control over the system, i.e., it is granular. This is significant when compared to a windows operating system where a user can control only what Microsoft allows them to. Everything in Linux, both at the minuscule and macro level, is controlled by the terminal. Additionally, scripting is simple and effective for any scripting language in Linux.

- Most Hacking Tools Are Written for Linux

A majority of tools used in hacking are written explicitly for Linux. There are some like Nmap, or Metasploit that can be available for the Windows platform, but still, not all their capabilities can be ported to windows. They offer limited functionalities as compared to when they are on the Linux platform.

4. The Future Belongs to Linux/Unix

Over the years, you may have witnessed that windows is slowing down and even stagnating in some departments. Since the advent of the internet, Linux/Unix has and is still the choice operating system for web servers primarily because of its reliability, robustness, and stability. To date,

almost two-thirds of web servers utilize Linux operating systems. Examples of uses of the Linux kernel are Citrix applications, VMware, embedded systems in switches and routers, mobile devices, and so on. It has been said that the future of computing is with mobile devices, including but not limited to phones and tablets. Android, which is used in most phones, is Linux, while iOS is a Unix kernel. It is, therefore, difficult to see how the future is not Linux/Unix. Microsoft Windows commands a meager market share of around 7 percent. The rest of the market is either Linux or Unix. In summary, the future lies with Linux/Unix.

What Is Kali Linux?

Kali Linux is a distribution of the Debian family. It was designed and developed solely for Security Auditing and Penetration Testing. The distro comprises hundreds of tools and utilities that are focused on information security tasks, which may include Reverse Engineering, Penetration Testing, Computer Forensics, and Security research. Offensive Security is the information security organization behind Kali Linux. It developed, funds, and maintains Kali Linux.
This distribution was initially launched in March 2013 to be a total rebuild of BackTrack Linux, top to bottom. It adhered to Debian development standards a hundred percent. Kali Linux boasts of over 600 tools that can be used for penetration testing. After a thorough review of BackTrack Linux, some tools that did not work or were in duplicate were eliminated. From the Kali Tools site, lets us look at some of the details:

Kali Linux will always be free: Just like BackTrack, kali is free of charge and always will be. This implies that you are not going to pay for it at any time now or in the future.

It is Open source: The source code Kali Linux uses is available to everyone that wants to improve, modify, or rebuild packages to adapt them to their specific requirements.

Kali complies with FHS: The distribution follows the Filesystem Hierarchy Standard. The standards help the users of Linux to locate support files, binaries, libraries, and so on seamlessly.

Support a wide range of wireless devices: The system has been designed with multiple platforms to support wireless interfaces. It can run on a wide range of hardware

The kernel is customized and is patched for injection: Latest injection patches are included in the Kali Linux kernel.

It was created in an environment that is secure: The team tasked with the development of Kali is a small group of individuals that are trusted to commit packages and interact with the repositories. Secure protocols are used in these processes.

GPG signed repositories and packages: All packages in Kali Linux are usually signed by the individual developers. It is these developers that are responsible for the packages. The packages are subsequently signed by the repositories as well.

Support for Multiple languages: Most penetration tools and utilities are often written in English. Kali, notwithstanding, offers true multilingual support. Users are therefore able to operate in their native languages. Isn't that a great thing!

Kali is easily customizable: All users can modify Kali to their requirements and preferences.

Kali Linux supports both ARMHF and ARMEL : The Kali Linux ARM supports fully working installations for both ARMEL and ARMHF systems. Kali Linux is available on a wide range of ARM devices and has ARM repositories integrated with the mainline distribution, so tools for ARM are updated in conjunction with the rest of the distribution.

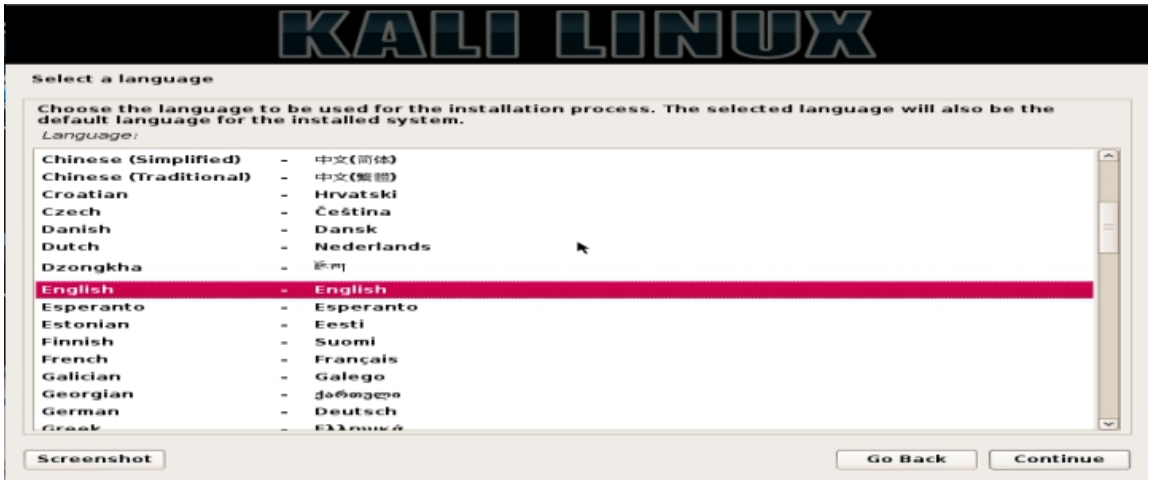
Downloading Kali Linux

Before I take you down the road towards being a hacker, you will first be required to download and install Kali Linux on your computer. This is the distribution of Linux that we shall be using throughout this book. That can be done from https://www.kali.org/ . Navigate to the home page and hit the Downloads link located at the top of the page. It is important that the right download is selected.

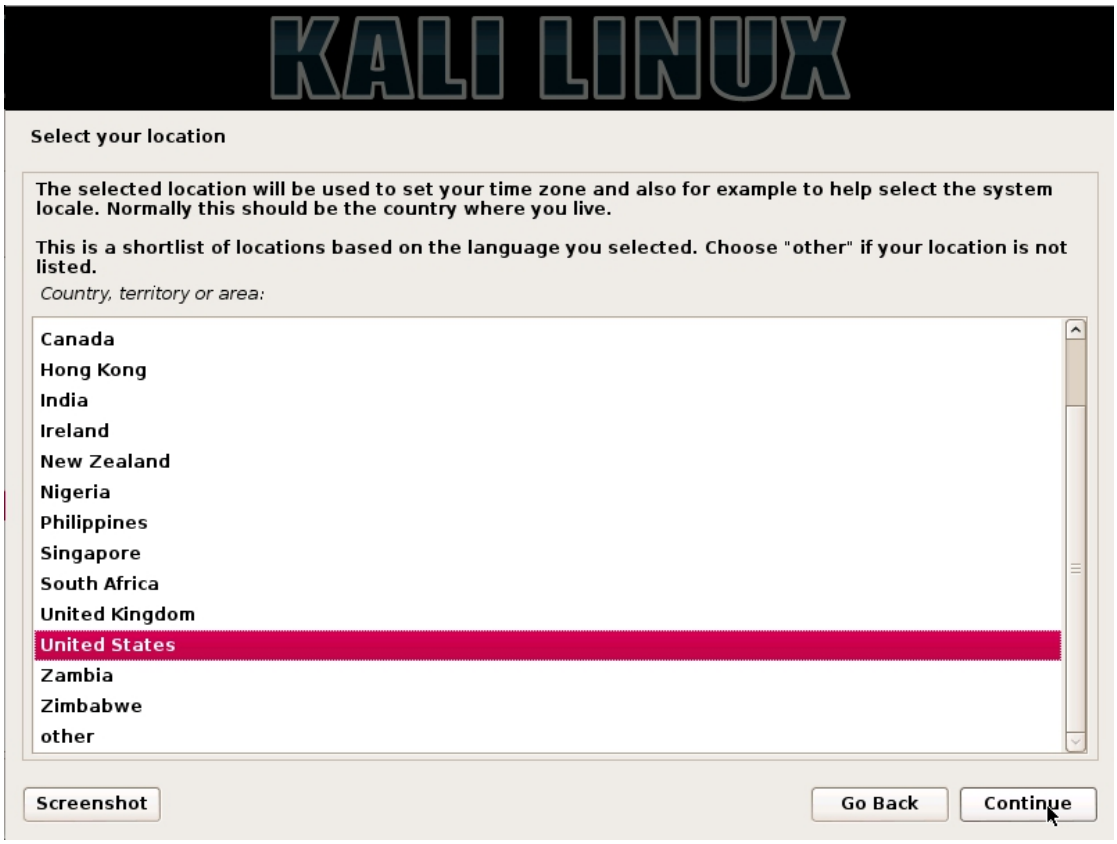
The procedure of Installing Kali and Setting Up Kali Linux

To start us off, you will need to boot using your preferred medium of installation. You should be greeted with the Kali Boot screen, as shown below. Choose either Text-Mode or Graphical install. In this example, we chose a GUI install.

Choose your language preference together with your country location. Also, you will be prompted to select your preferred keyboard layout.



Select your location, geographic that is.



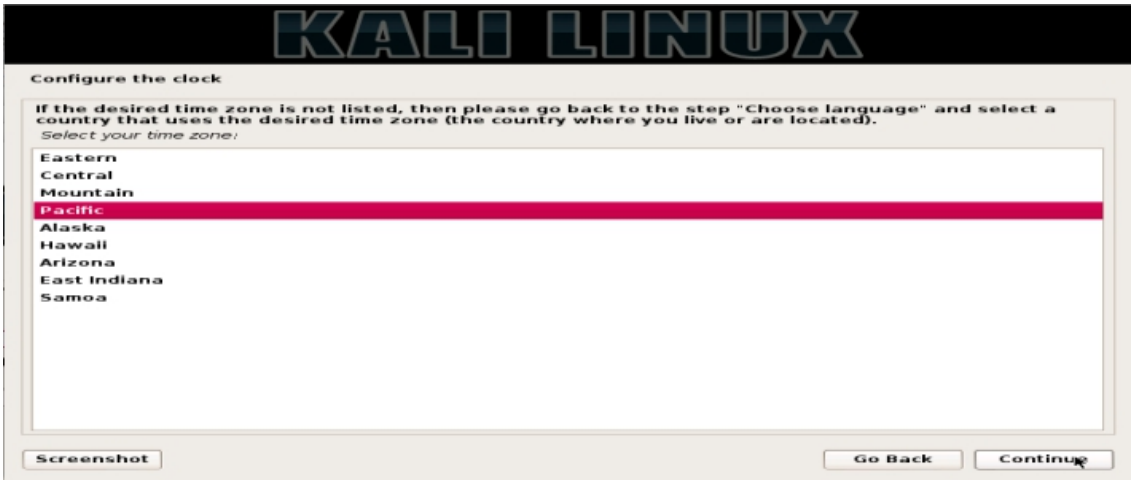
The installer will copy the image to your hard disk, probe your network interfaces, and then prompt you to enter a hostname for your system. In the example below, we've entered "kali" as our hostname.

You may optionally provide a default domain name for this system to use.

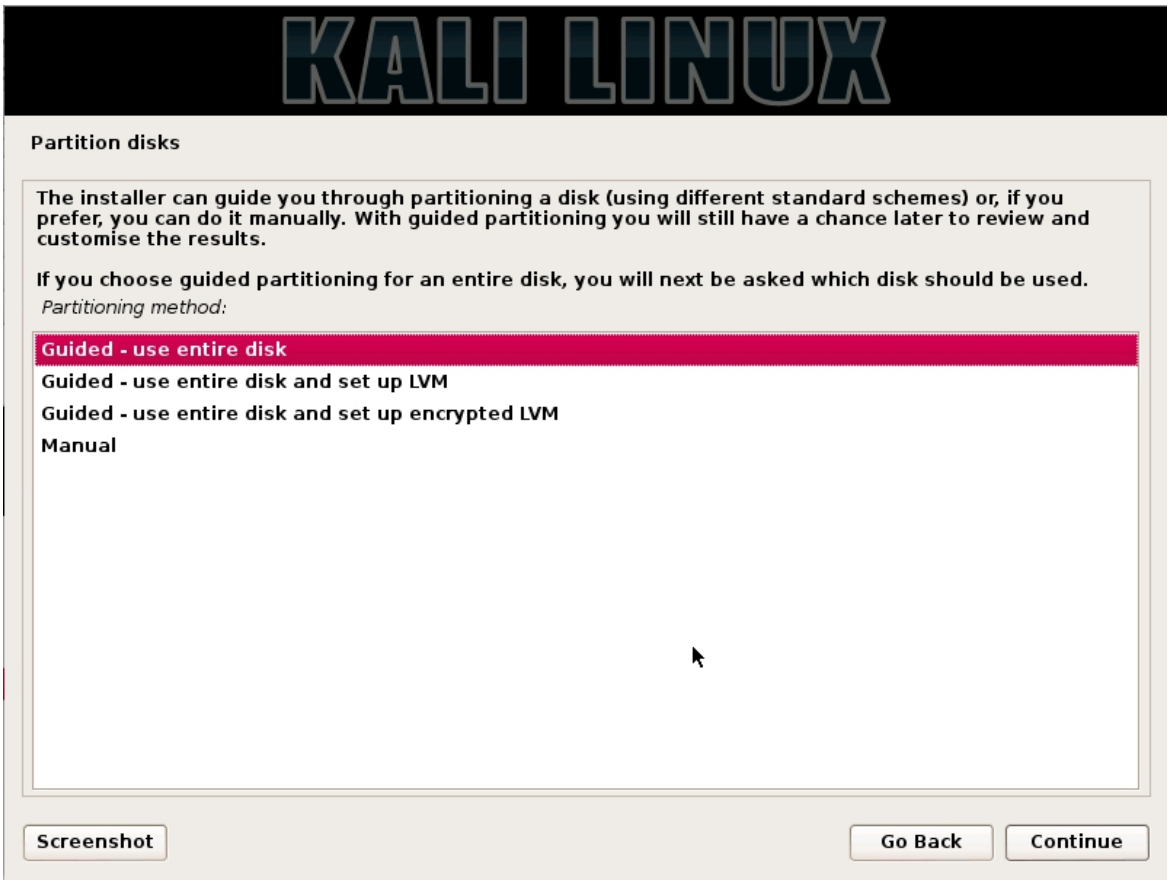
Next, provide a full name for a non-root user for the system.

A default user ID will be created, based on the full name you provided. You can change this if you like.

After that, pick an appropriate time zone.

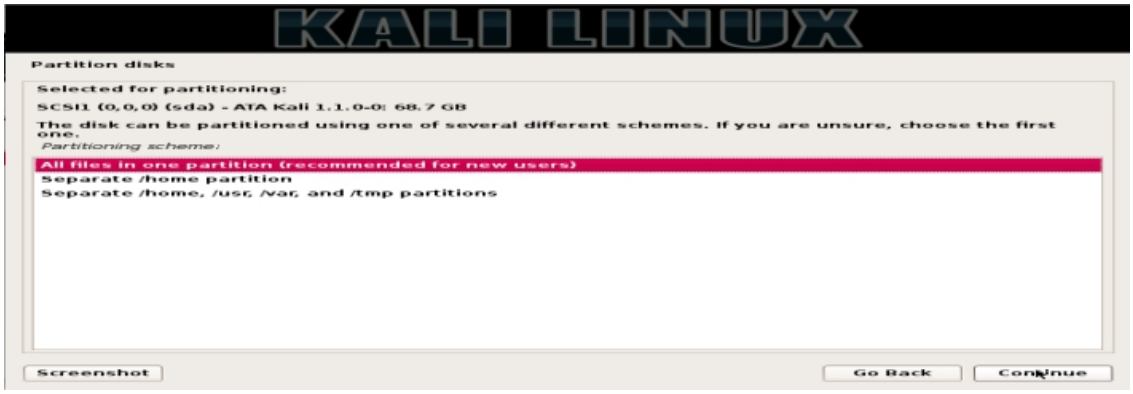


Next, you will see something similar to the picture below. Select appropriately.

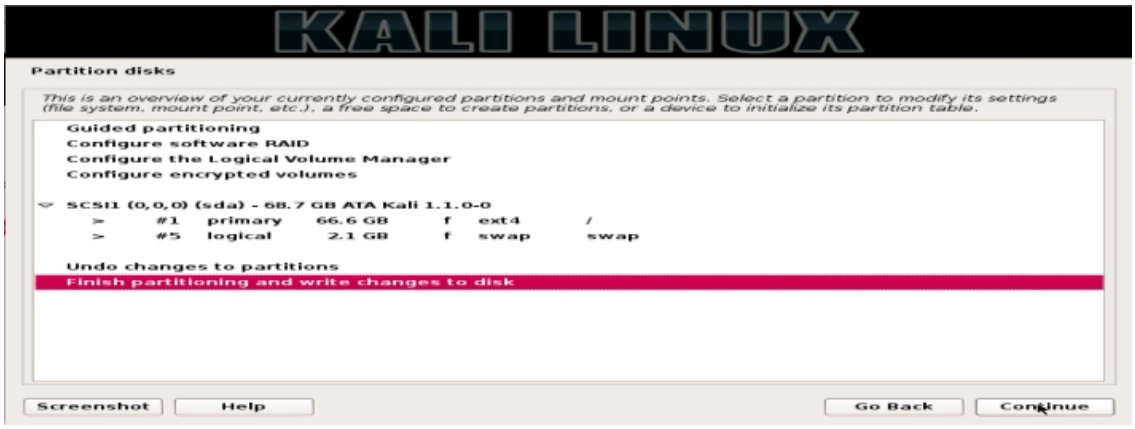


Choose the disk you want to be partitioned.

Continue to the next step below, selecting a choice depending on your needs.

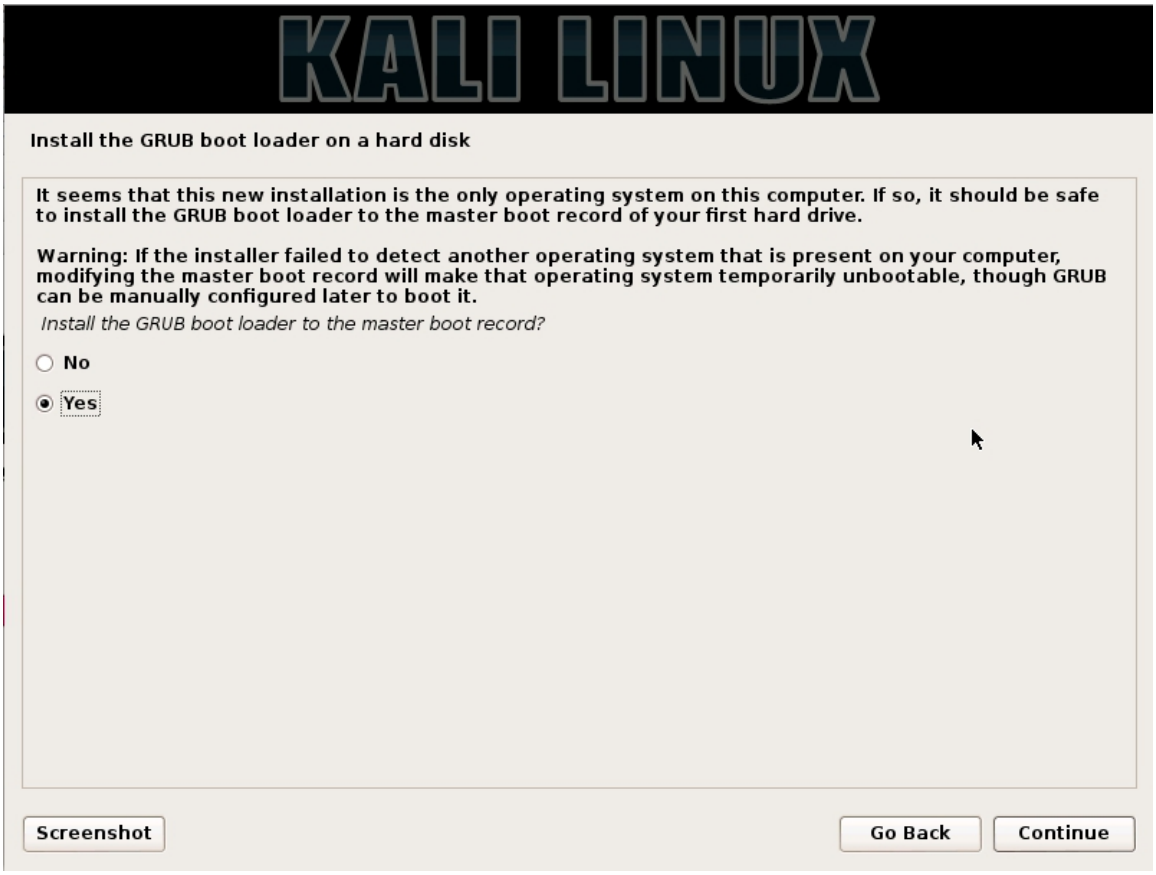


Next, hit the Continue button.



The next step requires you to carry out a configuration of network mirrors.

Next, install GRUB.



Finally, click Continue to reboot into your new Kali installation.


KALI LINUX
Finish the installation
Installation complete
Installation is complete, so it is time to boot into your new system. Make sure to remove the installation media (CD-ROM, floppies), so that you boot into the new system rather than restarting the installation.
Screenshot
Go Back
Continue

Chapter 4: Basics of Kali

Introduction

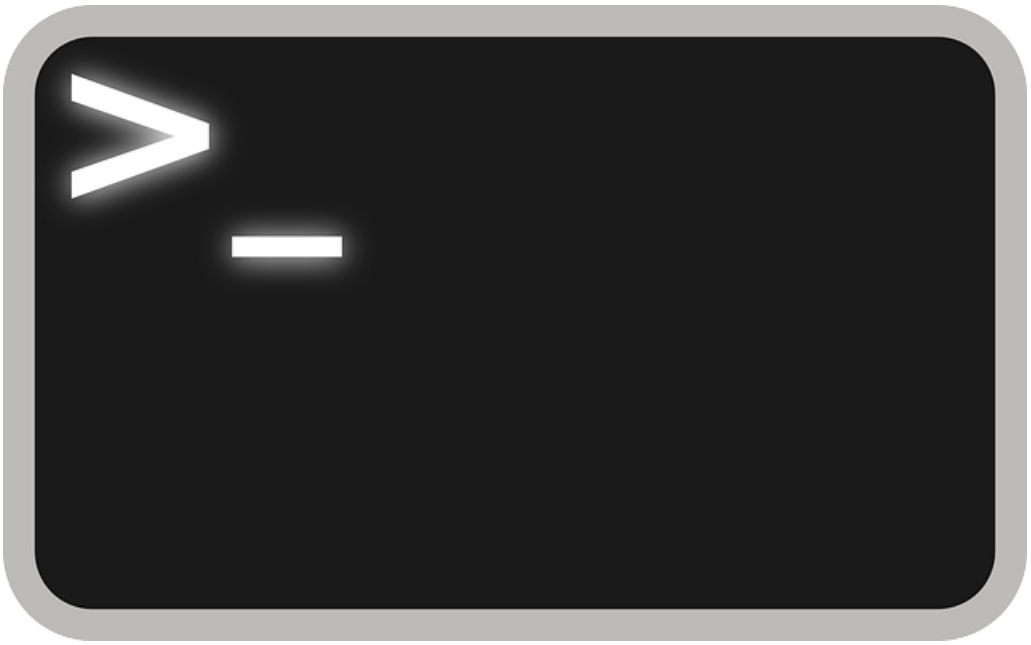
We have said previously that Kali Linux is a Debian based distribution for Ethical Hackers, Penetration Testers, Security Researchers and Enthusiasts. It is a stable, updated, enterprise-ready, open-source, and well-maintained distribution by Offensive Security. Kali Linux default desktop environment is GNOME, but it also offers a variety of other desktop environments including KDE, MATE, LXDE, and others. It can be installed on various types of systems, including laptops, Servers, ARM devices, and Cloud. It also has a portable version for android devices called NetHunter, which can be used within the Android operating system and comes with pre-installed tools and scripts that offer portability while doing security auditing or penetration testing.

Why You Should Use Kali Linux

As we have said before Kali Linux comes with just about every tool pre-installed that can be used for any of the above purposes. It is for this reason that Security Auditors, Forensics Investigators, Penetration Testers, and Researchers prefer it.

Kali can be used in the breaking of WiFi networks, to hack websites and networks, to run Open Source Intelligence on an entity among others. Kali Linux possesses tools that can be used for forensic investigation besides ethical hacking. This is becoming an equally essential branch of security that primarily collects evidence, analyze it, and uses the results to backtrack Cyber Criminals. Forensic Investigation makes it possible to locate and eradicate malicious effects emanating from malicious activities. It also comes in handy in the calculation and management of loss that occurs after a Cyber Attack. A key feature in Kali is the stealth Live mode mostly used in forensics and that it does not leave traces (fingerprints and footprints) on a host's system.

The Terminal



The very initial step in using Kali is to open the terminal, which is the command-line interface we'll use in this book. In Kali Linux, you'll find the icon for the terminal at the bottom of the desktop. Double-click this icon to open the terminal or press CTRLALTT. The terminal opens the command line environment, known as the shell, which enables you to run commands on the underlying operating systems and write scripts. Although Linux has many different shell environments, the most popular is the bash shell, which is also the default shell in Kali and many other Linux distributions. To change your password, you can use the command passwd.

Basic Commands in Linux

To begin, let's look at some basic commands that will help you get up and running in Linux.

- Finding Yourself with pwd

The command line in Linux does not always make it apparent which directory you're presently in unlike that in Windows or macOS. To navigate to a new directory, you usually need to know where you are currently. The present working directory command, pwd, returns your

location within the directory structure. Enter pwd in your terminal to see where you are:

```
kali >pwd
/root
```

In this case, Linux returned /root, telling me I'm in the root user's directory. And
because you logged in as root when you started Linux, you should be in the root user's directory, too, which is one level below the top of the filesystem structure (/). If you're in another directory, pwd will return that directory name instead.

- Checking Your Login with whoami

In Linux, the one "all-powerful" superuser or system administrator is called root, and it has all the system privileges needed to add users, change passwords, change privileges, and so on. Of course, you do not want just anyone to have the ability to make such changes; you want someone who can be trusted and has proper knowledge of the operating system. As a hacker, you usually need to have all those privileges to run the programs and commands you need, so you may want to log in as root. A Linux user can see which user they are logged in as using the "whoami" command as below:

```
kali >whoami
root
```

Here, the user is logged in as root.

- Navigating the Linux Filesystem

Navigating the filesystem from the terminal is an essential Linux skill. To get anything done, you need to be able to move around to find applications, files, and directories located in other directories. In a GUI-based system, you can visually see the directories, but when you're using the command-line interface, the structure is entirely text-based, and navigating the filesystem means using some commands.

- Changing Directories with cd

To change directories from the terminal, use the change directory command, cd. For example, here's how to change to the /etc. directory used to store configuration files:

```
kali >cd /etc
root@kali:/etc#
```

The prompt changes to root@kali:/etc, indicating that we're in the /etc. directory. We can confirm this by entering pwd

```
root@kali:/etc# pwd
/etc
```

To move up one level in the file structure (toward the root of the file structure, or /), we use cd followed by double dots (..), as shown here:

```
root@kali:/etc# cd ..
root@kali:/# pwd
/
root@kali:/#
```

This moves us up one level from /etc. to the /root directory, but you can move up as many levels as you need. Just use the same number of double dot pairs as the number of levels you want to move:

- You would use .. to move up one level.
- You would use .. .. to move up two levels.
- You would use .. .. .. to move up three levels, and so on.

So, for example, to move up two levels, enter cd followed by two sets of double dots with a space in between:

```
kali >cd .. ..
```

You can also move up to the root level in the file structure from anywhere by entering cd /, where / represents the root of the filesystem.

- Listing the Contents of a Directory with ls

To see the contents of a directory (the files and subdirectories), we can use the ls (list) command. This is very similar to the dir command in Windows.

```
kali >ls
bin  initrd.img   media   run   var
boot   initrd.img.old   mnt   sbin   vmlinuz
dev   lib      opt  srv    vmlinuz.old
etc  lib64     proc    tmp
home    lost+found   root   usr
```

```
kali >ls
bin   initrd.img      media      run      var
boot   initrd.img.old   mnt       sbin     vmlinuz
dev   lib           opt        srv      vmlinuz.old
etc   lib64         proc       tmp
home   lost+found     root       usr
```

This command lists both the files and directories contained in the directory. You can also use this command on any particular directory, not just the one you are currently in, by listing the directory name after the command; for example, ls /etc. shows what's in the /etc. directory. To get more information about the files and directories, such as their permissions, owner, size, and when they were last modified, you can add the -l switch after ls (the l stands for long). This is often referred to as long listing. See the example below:

```
kali >ls -l
total 84
drw-r--r--    1    root     root     4096    Dec    5  11:15    bin
drw-r--r--    2    root     root     4096    Dec    5  11:15    boot
drw-r--r--    3    root     root     4096    Dec    9  13:10    dev
drw-r--r--   18    root     root     4096    Dec    9  13:43    etc
--snip--
drw-r--r--    1    root     root     4096    Dec    5  11:15    var
```

- Getting Help

Nearly every command, application, or utility has a dedicated help file in Linux that guides its use. For instance, if I needed help using the best wireless cracking tool, aircrack-ng, I could type the aircrack-ng command followed by the --help command:

kali >aircrack-ng --help

Note the double dash here. The convention in Linux is to use a double dash (--) before word options, such as help, and a single dash (-) before single letter
options, such as –h. When you enter this command, you should see a short description of the tool and guidance on how to use it. In some cases, you can use either -h or -? to get to the help file. For instance, if I needed help using the hacker's best port scanning tool, Nmap, I would enter the following:

kali >nmap -h

Unfortunately, although many applications support all three options, there is no guarantee of the application you are using will. So, if one option refuses to work, please try another.

Finding Files

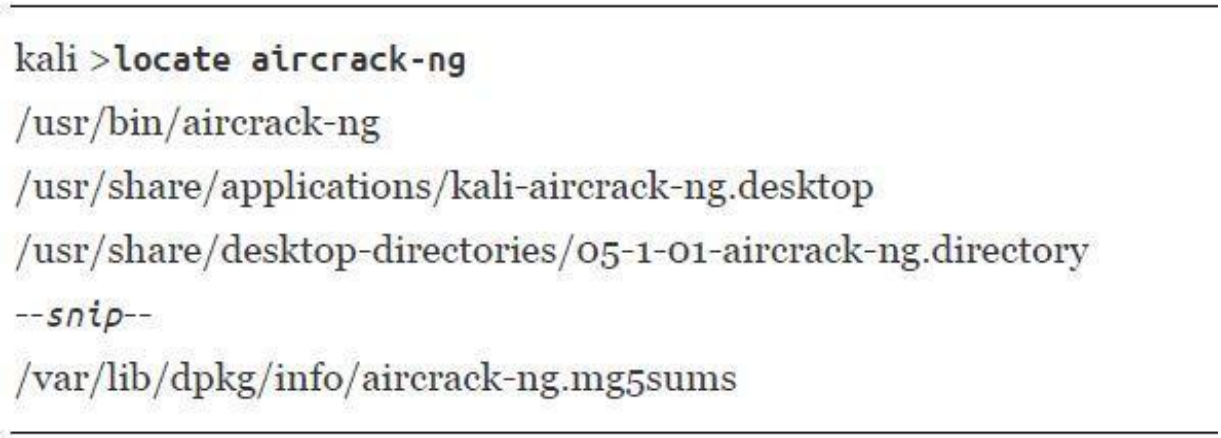
Until you become familiar with Linux, it can be frustrating to find your way around, but knowledge of a few basic commands and techniques will go a long way toward making the command line much friendlier. The following commands help you locate things from the terminal.

- Searching with locate

Probably the easiest command to use is locate. Followed by a keyword denoting what it is you want to find; this command will go through your entire filesystem and locate every occurrence of that word. To look for aircrack-ng, for example, enter the following:

kali >locate aircrack-ng
/usr/bin/aircrackng
/usr/share/applications/kaliaircrackng.desktop
/usr/share/desktopdirectories/05-1-01aircrack-ng.directory
--snip--
/var/lib/dpkg/info/aircrack-ng.mg5sums

A screenshot showing the output of the locate command looks like this;



The locate command is not perfect, however. Sometimes the results of locate can be overwhelming, giving you too much information. Also, locate uses a database that is usually only updated once a day, so if you just created a file a few minutes or a few hours ago, it might not appear in this list until the next day. It's worth knowing the disadvantages of these basic commands so you can better decide when best to use each one.

- Finding Binaries with whereis

If you're looking for a binary file, you can use the whereis command to locate it. This command returns not only the location of the binary but also its source and man page if they are available. Here's an example:

kali >whereis aircrack-ng
aircrack-ng:/usr/bin/aircrack-ng /usr/share/man/man1/aircrack-ng.1.gz

A screenshot showing the output of the whereis command looks like this;

- Finding Binaries in the PATH Variable with which

The which command is even more specific: it only returns the location of the binaries in the PATH variable in Linux. For example, when I enter aircrack-ng on the command line, the operating system looks to the PATH variable to see in which directories it should look for aircrackng:

kali >which aircrack-ng
/usr/bin/aircrack-ng

Here, which was able to find a single binary file in the directories listed in the PATH variable. At a minimum, these directories usually include /usr/bin, but may consist of/usr/sbin and maybe a few others.

- Performing More Powerful Searches with find

The find command is the most powerful and flexible of the searching utilities. It is capable of beginning your search in any designated directory and looking for several different parameters, including, of course, the filename but also the date of creation or modification, the owner, the group, permissions, and the size.
Here is the basic syntax for find:

find directory options expression

- Filtering with grep

Very often, when using the command line, you may want to search for a particular keyword. For this, you can use the grep command as a filter to

search for keywords. The grep command is often used when output is piped from one command to another.

A screenshot showing the output of the grep command looks like this;

```
kali >ps aux | grep apache2
root  4851 0.2 0.7 37548  7668 ?  Ss  10:14  0:00  /usr/sbin/apache2 -k start
root  4906 0.0 0.4 37572  4228 ?  S   10:14  0:00  /usr/sbin/apache2 -k start
root  4910 0.0 0.4 37572  4228 ?  Ss  10:14  0:00  /usr/sbin/apache2 -k start
--snip--
```

In the above example, the command will display all the services that are running and then pipe that output to grep. What grep does is it will search the received output for the keyword we asked it to look for. In our case, the keyword is apache2. Grep will go ahead and output only the relevant results. This command saves time.

Modify Files and Directories

After finding the directories and files you were looking for, you may need to carry out several operations on them. We are going to learn the creation of directories and files, copy files, rename files, plus delete the files and directories.

- Creating Files

There are many ways to create files in Linux, but for now, we will look at two simple methods. The first is cat, which is short for concatenate, meaning to combine pieces (not a reference to your favorite domesticated feline). The cat command is generally used for displaying the contents of a file, but it can also be used to create small files. For creating bigger files, it's better to enter the code in a text editor such as vim, emacs, Leafpad, gedit, or kate and then save it as a file.

- Concatenation with cat

The cat command followed by a filename will display the contents of that file, but to create a file, we follow the cat command with a redirect,

denoted with the > symbol, and a name for the file we want to create. Here is an example:

kali >cat > kalilinux
Hacking with Kali Linux!

- File Creation with touch

The second command for file creation is touch. This command was initially developed so a user could touch a file to change some of its details, such as the date it was created or modified. However, if the file does not already exist, this command creates that file by default. Let's create newfile using the touch command:

kali >touch newfile

Now when I then use ls –l to see the long list of the directory, I see that a new file has been created named newfile. Note that its size is 0 because there is no content in newfile.

- Creating a Directory

The command for creating a directory in Linux is mkdir, a contraction of make directory. To create a directory named newdirectory, enter the following command:

```
kali >mkdir newdirectory
```

To navigate to this newly created directory, do enter this:

```
kali >cd newdirectory
```

- Copying a File

To copy files, we use the cp command. This creates a duplicate of the file in the new location and leaves the old one in place. Here, we are going to create the file oldfile in the root directory with touch and copy it to /root/newdirectory, renaming it in the process and leaving the original oldfile in place:

```
kali >touch oldfile
kali >cp oldfile  /root/newdirectory/newfile
```

Renaming the file is optional and is done simply by adding the name you want to give it to the end of the directory path. If you don't rename the file when you copy it, the file will retain the original name by default. When we then navigate to newdirectory, we see that there is an exact copy of oldfile called newfile:

kali >cd newdirectory
kali >ls
newfile oldfile

- Renaming a File

Unfortunately, Linux doesn't have a command intended solely for renaming a file, as Windows and some other operating systems do, but it does have the mv (move) command. The mv command can be used to move a file or directory to a new location or to give an existing file a new name. To rename newfile to newfile2, you would enter the following:

kali >mv newfile newfile2
kali >ls
oldfile newfile2

Below is a screenshot of the same.

```
kali >mv newfile newfile2
kali >ls
oldfile newfile2
```

Now when you list (ls) that directory, you see newfile2 but not newfile, because it has been renamed. You can do the same with directories.

Removing a File
To remove a file, you can use the rm command, like so:

kali >rm newfile2

If you now do a long listing on the directory, you can confirm that the file has been removed.

Removing a Directory
The command for removing a directory is similar to the rm command for removing files but with dir (for directory) appended, like so:

kali >rmdir newdirectory
rmdir:failed to remove 'newdirectory': Directory not empty

Below is a screenshot of the same.

```
kali >rmdir newdirectory
rmdir:failed to remove 'newdirectory': Directory not empty
```

It is important to note that rmdir will not remove a directory that is not empty but will give you a warning message that the “directory is not empty,” as you can see in this example. You must first remove all the contents of the directory before removing it. This is to stop you from accidentally deleting objects you did not intend to delete. If you do want to

remove a directory and its content all in one go, you can use the -r switch after rm, as shown below:

kali >rm -r newdirectory

Just a word of caution, though: be wary of using the -r option with rm, at least at first, because it is straightforward to remove valuable files and directories by mistake. Using rm -r in your home directory, for instance, would delete every file and directory there, that is certainly not what you were intending.

Searching for tools/packages
Before you download a software package, you can check whether the package you need is available from your repository, which is a place where your operating system stores information. The apt tool has a search function that can check whether the package is available. The syntax is straightforward:

apt-cache search keyword

The screenshot has been attached for your reference.

---

apt-cache search *keyword*

Note that we use the apt-cache command to search the apt cache or the place it stores the package names. So, if you were searching for the intrusion detection system Snort, for example, you would enter the command shown below.

kali >apt-cache search snort
fwsnort – Snort-to-iptables rule translator
ippl - IP protocols logger
--snip--
snort – flexible Network Intrusion Detection System
Snort - common - flexible Network Intrusion Detection System - common files
--snip--

Take note of the exact spacings between the command from the screenshot below.

```
kali >apt-cache search snort
fwsnort - Snort-to-iptables rule translator
ippl - IP protocols logger
--snip--
snort - flexible Network Intrusion Detection System
snort-common - flexible Network Intrusion Detection System - common files
--snip--
```

As you can see, many files have the keyword snort in them, but near the middle of the output, we see snort – flexible Network Intrusion Detection System. That is what we are looking for.

Adding Softwares

Now that you know the snort package exists in your repository, you can use apt-get to download the software. To install a piece of software from your operating system's default repository in the terminal, use the apt-get command, followed by the keyword install and then the name of the package you want to install. The syntax looks like this:

apt-get install packagename

Let us try this out by installing Snort on your system. Enter apt-get install snort as a command statement, as shown below.

kali >apt-get install snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
snort-doc
The following NEW packages will be installed:
snort
--snip—
Install these packages without verification [Y/n]?

Here is a screenshot of the same.

```
kali >apt-get install snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
snort-doc
The following NEW packages will be installed:
snort
--snip--
Install these packages without verification [Y/n]?
```

The output you see tells you what is being installed. If everything looks correct, go
ahead and enter y when prompted, and your software installation will proceed.

Removing Softwares
When removing software, use apt-get with the remove option, followed by the name of the software to remove. An example is listed below.

kali >apt-get remove snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
libdaq0 libprelude2 oinkmaster snort-common-libraries snort-rules-default
--snip--
Do you want to continue [Y/n]?

```
kali >apt-get remove snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  libdaq0 libprelude2 oinkmaster snort-common-libraries snort-rules-default
--snip--
Do you want to continue [Y/n]?
```

Again, you will see the tasks being done in real-time, and you will be asked whether you want to continue. You can enter y to uninstall, but you might want to keep Snort since we will be using it again. The remove command does not remove the configuration files, which means you can reinstall the same package in the future without reconfiguring. If you do want to remove the configuration files at the same time as the package, you can use the purge option, as shown below.

```
kali >apt-get purge snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer
required:
    libdaq0 libprelude2 oinkmaster snort-common-libraries snort-rules-
default
--snip--
Do you want to continue [Y/N]?
```

```
kali >apt-get purge  snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libdaq0 libprelude2 oinkmaster snort-common-libraries snort-rules-default
--snip--
Do you want to continue [Y/n]?
```

Enter Y at the prompt to continue the purge of the software package and the configuration files. To keep things small and modular, many Linux packages are broken into software units that many different programs might use. When you installed Snort, you installed several dependencies or libraries with it that Snort requires so that it can run. Now that you are removing Snort, those other libraries or dependencies are no longer needed, so they are removed, too.

Updating Packages

Software repositories will be periodically updated with new software or new versions of existing software. These updates do not reach you automatically, so you need to request them to apply these updates to your system. Updating is different from upgrading: updating updates the list of packages available for download from the repository, whereas upgrading will upgrade the package to the latest version in the repository. You can update your system by entering the apt-get command followed by the keyword update. This will search through all the packages on your system and check whether updates are available. If so, the updates will be downloaded. See the example below.

```
kali >apt-get update
Get:1 http://mirrors.ocf.berkeley.edu/kali kali-rolling InRelease [30.5kb]
Get:2 http://mirrors.ocf.berkeley.edu/kali kali-rolling/main amd64
Packages
[14.9MB]
Get:3 http://mirrors.ocf.berkeley.edu/kali kali-rolling non-free amd64
Packages
[163kb]
Get:4 http://mirrors.ocf.berkeley.edu/kali kali-rolling/contrib amd64
Packages [107kB]
Fetched 15.2 MB in 1min 4s (236 kB/s)
Reading package lists... Done
```

The list of available software in the repository on your system will be updated. If the update is successful, your terminal will state Reading package lists... Done, as you can see above. Note that the name of the repository and the values; time, size, and so on, might be different on your system.

Upgrading Packages

To upgrade the existing packages on your system, use apt-get upgrade. Because upgrading your packages may make changes to your software, you must be logged in as root or use the sudo command before entering apt-get upgrade. This command will upgrade every package on your system that apt knows about, meaning only those stored in the repository as shown below. Upgrading can be time-consuming, so you might not be able to use your system for a while.

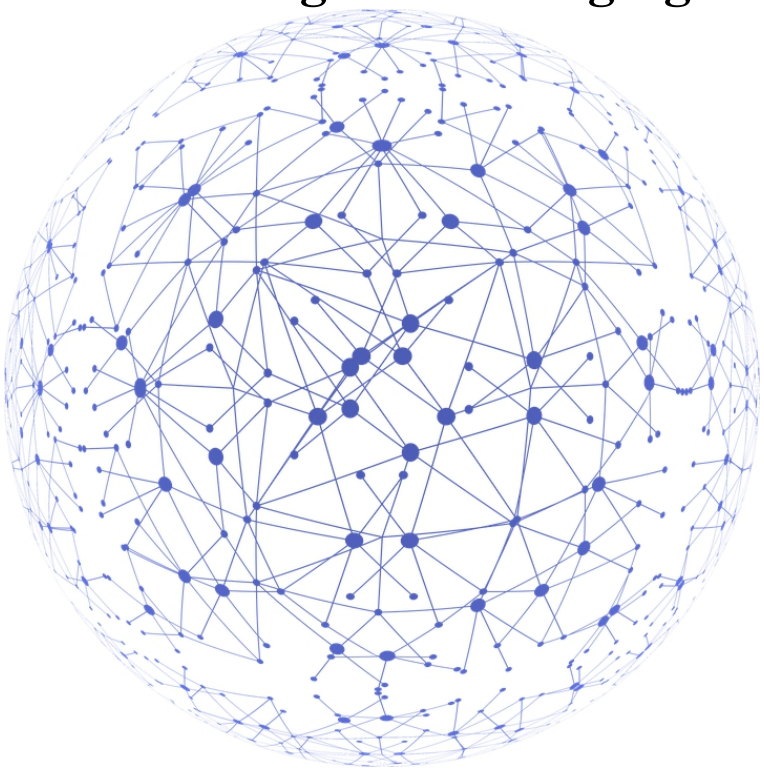
kali >apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Calculating upgrade... Done
The following packages were automatically installed and no longer required:
--Snip—
The following packages will be upgraded:
--snip—
1101 upgraded, 0 newly installed, 0 to remove and 318 not upgraded.
Need to get 827 MB of archives.
After this operation, 408 MB disk space will be freed.
Do you want to continue? [Y/n]

---

kali >**apt-get upgrade**
Reading package lists... Done
Building dependency tree... Done
Calculating upgrade... Done
The following packages were automatically installed and no longer required:
--snip--
The following packages will be upgraded:
--snip--
1101 upgraded, 0 newly installed, 0 to remove and 318 not upgraded.
Need to get 827 MB of archives.
After this operation, 408 MB disk space will be freed.
Do you want to continue? [Y/n]

You should see in the output that your system estimates the amount of hard drive space necessary for the software package. Go ahead and enter Y if you want to continue and have enough hard drive space for the upgrade.

Chapter 5: Scanning and Managing Networks



Introduction

The ability to scan for and connect to other network devices from your system is crucial to becoming a successful hacker, and with wireless technologies like WiFi
and Bluetooth becoming the standard, finding and controlling WiFi and Bluetooth connections is vital. If someone can hack a wireless connection, they can gain entry to a device and access to confidential information. The first step, of course, is to learn how to find these devices. In this chapter, we are going to examine two of the most common wireless technologies in Linux: WiFi and Bluetooth.

Network Scanning

We say that it is the utilization of a computer network for purposes of collecting information about IT systems. We carry out scanning of networks primarily to help us do system maintenance or a security assessment. Hackers can also conduct a network scanning exercise before launching their attacks. The following are some of the reasons we scan networks:

- Identification of the available UDP and TCP network services that may be running on the targets.
- To get to understand the systems for filtering that are in between the targeted hosts and the user.
- Discover the operating systems that are being used through the assessment of their IP responses.
- Analyze a particular host that is being targeted for its TCP sequence number predictability to enable the prediction of TCP spoofing and the attack sequence.

Network scanning comprises of two key aspects: Vulnerability scanning and network port scanning. The latter denotes a way of sending data packets through a network over to a systems' specific port numbers. The goal is to discover network services that are present in that particular system. It is an excellent way for troubleshooting issues that a given system has. That way, the problems can be dealt with so that the system is secure. For us to discover known vulnerabilities present in network systems, a method known as vulnerability scanning is used. Through it, we can identify weak spots both in the operating system and the application software. It is these weak points that are usually used to compromise computing systems.

Both vulnerability scanning and network port scanning can be said to be techniques that are used in information gathering. On the flip side, they can be a prelude to an attack when they are put to use by anonymous entities. Such entities usually have malicious intentions. Inverse mapping is another technique for network scanning. It is useful when it comes to collecting IP addresses that are not mapped to live hosts. By doing so, it will be aiding in the focusing attention on addresses that are worth focusing on, that is, those that are feasible. There are three stages in which information gathering can be accomplished.

i. The footprinting stage,
ii. The scanning stage, and
iii. The enumeration stage.

This, therefore, implies that network scanning is among the crucial steps an attacker needs to be able to gather information.

- **Network Scanning with ifconfig**

The ifconfig command is one of the essential tools that can be used for examining and interacting with active network interfaces. You can use it to query your active network connections by simply entering ifconfig in the terminal.

- **Scanning Wireless Networks with iwconfig**

If you have a wireless adapter, you can use the iwconfig command to gather crucial information for wireless hacking such as the adapter's IP address, its MAC address, what mode it's in, and more. The information you can glean from this command is particularly important when you're using wireless hacking tools like aircrackng.

Changing Your Network Information

Being able to change your IP address and other network information is a useful skill because it will help you access other networks while appearing as a trusted device on those networks. For example, in a denial of service (DoS) attack, you can spoof your IP so that that the attack appears to come from another source, thus helping you evade IP capture during forensic analysis. This is a relatively simple task in Linux, and it's done with the ifconfig command.

- **Changing Your IP Address**

To change your IP address, enter ifconfig followed by the interface you want to reassign and the new IP address you want to be assigned to that interface. For example, to assign the IP address 192.168.181.115 to interface eth0, you would enter the following:

Kali >ifconfig eth0 192.168.181.115
kali >

When you do this correctly, Linux will go back to the command prompt and say nothing. This is a good thing! Then, when you again check your network connections with ifconfig, you should see that your IP address has changed to the new IP address you just assigned.

- **Changing Your Network Mask and Broadcast Address**

You can also change your network mask (netmask) and broadcast address with the ifconfig command. For instance, if you want to assign that same eth0 interface with a netmask of 255.255.0.0 and a broadcast address of 192.168.1.255, you would enter the following:

Kali >ifconfig eth0 192.168.181.115 netmask 255.255.0.0 broadcast

192.168.1.255
kali >

Once again, if you've done everything correctly, Linux responds with a new command prompt. Now enter ifconfig again to verify that each of the parameters has been changed accordingly.

- **Spoofing Your MAC Address**

You can also use ifconfig to change your MAC address. The MAC address is globally unique and is often used as a security measure to keep hackers out of networks —or to trace them. Changing your MAC address to spoof a different MAC address is almost trivial and neutralizes those security measures. Thus, it's an instrumental technique for bypassing network access controls. To spoof your MAC address, use the ifconfig command's down option to take down the interface (eth0 in this case). Then enter the ifconfig command followed by the interface name (hw for hardware, ether for Ethernet) and the new spoofed MAC address. Finally, bring the interface back up with the up option for the change to take place.

- **IP Addresses Assignment**

Linux has a Dynamic Host Configuration Protocol (DHCP) server that runs a daemon, a process that runs in the background, called DHCPD, or the DHCP daemon. The DHCP server will carry out the assignment of IP addresses to all of the systems that are located on the subnet. It also keeps a log of which IP address is allocated to which machine at any one time. This makes it an excellent resource for forensic analysts to trace hackers after an attack. For that reason, it's useful to understand how the DHCP server works. Usually, to connect to the internet from a LAN, you must have a DHCP-assigned IP.

Therefore, after setting a static IP address, you must return and get a new DHCP-assigned IP address. To do this, you can always reboot your system, but I will show you how to retrieve a new DHCP without having to shut your system down and restart it. To request an IP address from DHCP, all that is required is to call the DHCP server using dhclient followed by an interface that you wish to assign the address. The different Linux distros

use different DHCP clients. Kali, for instance, is based on Debian that uses dhclient.

Manipulating the Domain Name System (DNS)

Hackers can find a treasure trove of information on a target in its Domain Name
System (DNS). DNS is a critical component of the internet, and although it's designed to translate domain names to IP addresses, a hacker can use it to garner information on the target.

- **Examining DNS with dig**

DNS is the service that translates a domain name like google.com to the appropriate IP address. This way, your system knows how to get to it. Without DNS, it would mean that we would be required to remember the thousands of IP addresses that belong to the websites we visit frequently. Dig is one of the commands any aspiring hacker needs to know. It offers a way to gather DNS information about a target domain. The stored DNS information can be a crucial piece of early reconnaissance to obtain before attacking. This information could include the IP address of the target's nameserver (the server that translates the target's name to an IP address), the target's email server, and potentially any subdomains and IP addresses. You can also use the dig command to get information on email servers connected to a domain by adding the mx option (mx is short for mail exchange server). This information is critical for attacks on email systems.

- **Changing Your DNS Server**

In some cases, you may want to use another DNS server. To do so, you will edit a plaintext file named /etc/resolv.conf on the system. Open that file in a text editor. Then, on your command line, enter the precise name of your editor followed by the location of the file and the filename.

Wi-Fi Networks

Firstly, let us look at WiFi. Before doing so, here is a small introduction to the various WiFi security protocols that usually are frequently used. The original, Wired Equivalent Privacy (WEP), was severely flawed and easily cracked. Its replacement, WiFi Protected Access (WPA), was a bit more secure. Finally, WPA2PSK, which is much more secure and uses a pre-shared key (PSK) that all users share, is now used by nearly all WiFi AP's (except enterprise WiFi).

- **Basic Wireless Commands**

ifconfig

To perform a network interface configuration in Unix-based operating systems, one needs ifconfig. It is an administration utility that is found in the system. Ifconfig has utilities that are utilized in the configuration, querying, and controlling of the parameters of the TCP/IP interface. As an interactive tool, ifconfig can be used to show settings of the network interface and analyze them.

In summary, ifconfig does the following:

- The command enables viewing the settings of a network.
- Carrying out enabling of a network Interface and also disabling it
- Network Interface IP address assigning
- Assigning network interfaces, a netmask
- Allocating a Broadcast to Network Interface

- Assigning an IP, Netmask, and Broadcast to Network Interface
- Changing MTU for a Network Interface
- Enabling and disabling Promiscuous Mode
- Addition and removal of New Alias to Network Interface
- Changing the MAC address of Network Interface

iwevent

This command displays Wireless Events received through the RTNetlink socket. Each line shows the specific Wireless Event which describes what has happened on the specified wireless interface. This command doesn't take any arguments.

iwlist

This command can be used for scanning wireless networks available and also for displaying any other information about the wireless networks which are not displayed when the iwconfig command is used. iwlist is utilized in the generation of wireless access points that are nearby together with their SSIDs and their MAC addresses.

iwspy

This command is used for monitoring nodes in a network. It can also be used for recording the link quality of the nodes.

ifrename

This command is used for renaming wireless network interfaces depending on multiple criteria that are static to allocate names consistently to each interface. The interface names usually are dynamic by default. This command helps users decide the name of the network interface. The command needs to be executed before bringing the interfaces up.

iwgetid

This is used in the reporting of the NWID, ESSID, or address of the access point of the wireless network presently being used. By default, iwgetid will display the devices' ESSID. Suppose that it is unavailable, it will output its NWID instead. The information reported is the same as the one shown by iwconfig. In comparison, it is easier to do integration in various scripts.

- **Detecting and Connecting to Bluetooth**

In recent times, nearly all gadgets, systems, and devices have inbuilt Bluetooth. The devices can be computers, iPods, smartphones, speakers, game controllers, keyboards, tablets, among others. The ability to break into Bluetooth networks can result in the compromising of the information on the device, assuming a devices' control, and acquisition of a platform to transmit privileges information from and to the device, among other things. We, therefore, need to understand how Bluetooth works if we are to exploit this technology. From this book, you will be able to acquire some basic knowledge that will come in handy during the scanning and connecting to Bluetooth devices in preparation for hacking them.

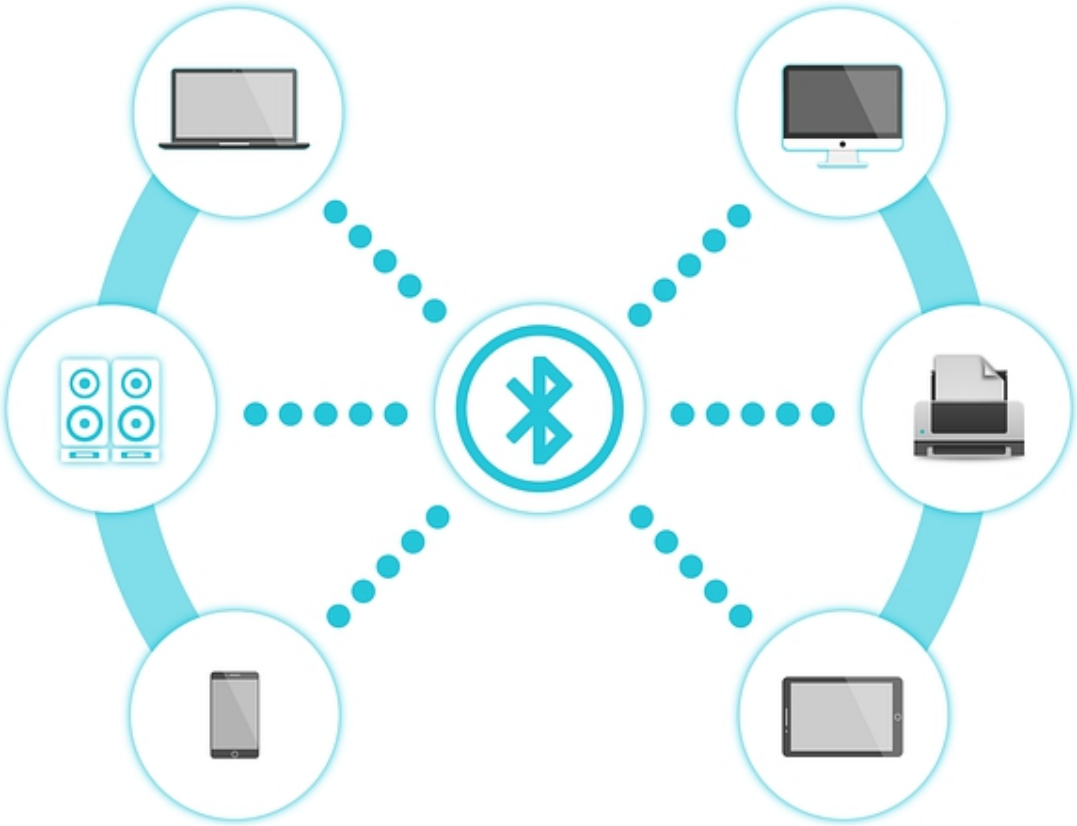
How Bluetooth Works

First, we can define Bluetooth as a wireless communication technology that enables devices to transmit voice or data wirelessly. This happens over a relatively short distance. This technology was meant to replace the ubiquitous cables that were being used to connect devices while still securing the communications across them. The process of joining two Bluetooth devices is known as pairing. Pretty much any two devices can pair if they are set to a discoverable mode. In the discoverable mode, a Bluetooth device will broadcast the following information about themselves:

- Technical information
- Name
- List of services
- Class

Upon pairing, two Bluetooth devices will exchange a link key. The devices will store the key to be used in the identification of the other device in future pairings. Every device has a unique identifier and usually a manufacturer-assigned name. These will be useful pieces of data when we want to identify and access a device.

Bluetooth Scanning and Reconnaissance



Linux has an implementation of the Bluetooth protocol stack called BlueZ that we are going to use to scan for Bluetooth signals. Most Linux distributions, including Kali Linux, have it as an inbuilt feature by default. BlueZ possesses utilities that can help us scan and manage Bluetooth capable devices. Examples of the utilities are outlined below:

- hciconfig; this is an equivalent of ifconfig in Linux, but made for Bluetooth capable devices.
- hcitool; this is a tool that we use to perform inquiries. The inquiries can be the device ID, name, class, or even its clock information. This helps the devices to work in sync.
- hcidump; sniffing of Bluetooth communications is carried out by this tool, it, therefore, gives us a chance to capture data that is being sent over the Bluetooth signal.

The first scanning and reconnaissance step with Bluetooth is to check whether the Bluetooth adapter on the system that we are using is

recognized and enabled so we can use it to scan for other devices.

Scanning for Bluetooth Devices with hcitool

Now that we know our adapter is up, we can use another tool in the BlueZ suite called hcitool, which is used to scan for other Bluetooth devices within range.
With the simple scan command, we can find out Bluetooth devices that are transmitting using their discover beacons. That is, the devices set to their discovery mode. Most of the tools for Bluetooth hacking you are likely to encounter will be using these commands in a script. You should be able to create your tools from these commands using Python script or even bash script.

Using the sdptool to Scanning for Services

The service discovery protocol, SDP as it is commonly known, is a protocol of Bluetooth that is used in the searching of Bluetooth services (Bluetooth is a suite of services), and, helpfully, BlueZ provides the sdptool tool for browsing a device for the services it offers. It is also important to note that the device does not have to be in discovery mode to be scanned. The syntax is as follows:

sdptool browse MACaddress

Seeing Whether the Devices Are Reachable with l2ping

Once we have gathered the MAC addresses of all nearby devices, we can send out pings to these devices, whether they are in discovery mode or not, to see whether they are in reach. This lets us know whether they are active and within range. To send out a ping, we use the l2ping command with the following syntax:

l2ping MACaddress

Summary

Wireless devices represent the future of connectivity and hacking. Linux has developed specialized commands for scanning and connecting to Wi-

Fi APs in the first step toward hacking those systems. The aircrack-ng suite of wireless hacking tools includes both airmon-ng and airodump-ng, which enable us to scan and gather vital information from in-range wireless devices. The BlueZ suite includes hciconfig, hcitool, and other tools capable of scanning and information gathering, which are necessary for hacking the Bluetooth devices within range. It also includes many other tools worth exploring.

Chapter 6: File and Directories Permissions

Introduction

Not every user of a single operating system should have the same level of access to files and directories. Like any professional or enterprise-level operating system, Linux has methods for securing file and directory access. This security system allows the system administrator, the root user, or the file owner to protect their files from unwanted access or tampering by granting select users' permissions to read, write, or execute files. For each file and directory, we can specify the permission status for the file's owner, for groups of users, and all other users. This is a necessity in a multiuser, enterprise-level operating system. The alternative would be quite chaotic. In this chapter, I will show you how to check for and change permissions on files and directories for select users, how to set default file and directory permissions, and how to set special permissions. Finally, you will see how a hacker's understanding of permissions might help them exploit a system.

Types of Users

As you know, in Linux, the root user is all-powerful. The root user can do just about anything on the system. Other users on the system have more limited capabilities and permissions and seldom have the access that the root user has. These other users are usually collected into groups that generally share a similar function. In a commercial entity, these groups

might be finance, engineering, sales, and so on. In an IT environment, these groups might include developers, network administrators, and database administrators. The idea is to put people with similar needs into a group that is granted relevant permissions; then each member of the group inherits the group permissions. This is primarily for the ease of administering permissions and, thus, security. The root user is part of the root group by default. Each new user on the system must be added to a group to inherit the permissions of that group.

Granting Permissions

Each file and directory must be allocated a particular level of permission for the different identities using it. The three levels of permission are listed hereunder:

r - Read permission. This grants permission only to open and view a file.
w - Write permission. This allows users to view and edit a file.
x - Execute permission. This will enable users to execute a file (This does not guarantee viewing or editing it).

In this way, the root user can grant users a level of permission depending on what they need the files for. When a file is created, typically the user who created it is the owner of the file, and the owning group is the user's current group. The owner of the file can grant various access privileges to it. Let us have a look at how to change permissions to pass ownership to individual users and groups.

1. Granting Ownership to an Individual User

To move ownership of a file to a different user so that they can control permissions, we can use the chown (or change owner) command:

kali >chown ❶ John ❷ /tmp/johnsfile

Here, we give the command, the name of the user we are giving ownership to, and then the location and name of the relevant file. This command grants the user account for John ❶ ownership of johnsfile ❷ .

2. Granting Ownership to a Group

To transfer ownership of a file from one group to another, we can use the chgrp (or change group) command. Hackers are often more likely to work alone than in groups, but it's not unheard of for several hackers or penetration testers work together on a project, and in that case, using groups is necessary. For instance, you might have a group of penetration testers and a group of security team members working on the same project. The penetration testers in this example are the root group, meaning they have all permissions and access. The root group needs access to the hacking tools, whereas the security folk only need access to defensive tools such as an intrusion detection system (IDS). Let's say the root group downloads and installs a program named newIDS; the root group will need to change the ownership to the security group so the security group can use it at will. To do so, the root group would enter the following command:

```
kali >chgrp ❶ security ❷ newIDS
```

This command passes the security group ❶ ownership of newIDS ❷ .

Now you need to know how to check whether these allocations have worked. You will do that by checking a file's permissions.

Checking and Changing Permissions

When you want to find out what file or directory permissions are granted to what users, the ls command can be used. It will, however, need to be suffixed by –l (long) switch that is used for displaying a directory's content in long format. The displayed list will contain the permissions. The syntax is as below:

```
kali >ls –l /usr/share/hashcat
```

The output will display, among other things, the permissions on the file. We have three sets of three characters, made of some combination of read (r), write (w), and execute (x), in that order. The first set represents the permissions of the owner; the second, those of the group; and the last, those of all other users.

Regardless of which set of three letters you are looking at if you see an r first, that user or group of users has permission to open and read that file or directory. A w as the middle letter means they can write to (modify) the file or directory, and an x at the end means they can execute (or run) the file or directory. If any r, w, or x is replaced with a dash (-), then the respective permission hasn't been given. Note that users can have permission to execute only either binaries or scripts.

Changing Permissions

We can use the Linux command chmod to change the permissions. Only a root user or the file's owner can change permissions. In this section, we use chmod to change permissions on hashcat.hcstat using two different methods. First, we use a numerical representation of permissions, and then we use a symbolic representation.

- **Changing Permissions with Decimal Notation**

We can use a shortcut to refer to permissions by using a single number to represent one rwx set of permissions. Like everything underlying the operating system, permissions are represented in binary, so ON and OFF switches are represented by 1 and 0, respectively. You can think of the rwx permissions as three ON/OFF switches, so when all permissions are granted, this equates to 111 in binary. A binary set like this is then easily represented as one digit by converting it into octal, an eight-digit number system that starts with 0 and ends with 7. An octal digit represents a set of three binary digits, meaning we can represent an entire rwx set with one digit.

- **Changing Permissions with UGO**

Although the numeric method is probably the most common method for changing permissions in Linux, some people find chmod's symbolic method more intuitive. Both methods work equally well, so find the one that suits you. The symbolic method is often referred to as the UGO syntax, which stands for the user (or owner), group, and others. UGO syntax is quite straightforward. Enter the chmod command and then the users you want to change permissions for, providing u for the user, g for group, or o for others, followed by one of three operators:

- Removes a permission
+ Adds a permission
= Sets a permission

After the operator, include the permission you want to add or remove (rwx) and, finally, the name of the file to apply it to. So, if you want to remove the write permission from the user that the file hashcat.hcstat belongs to; you could enter the following:

kali >chmod u-w hashcat.hcstat

This command says to remove (-) the write (w) permission from hashcat.hcstat for the user (u). Now when you check the permissions with ls –l again, you should see that the hashcat.hcstat file no longer has write permission for the user:

- Giving Root Execute Permission on a New Tool

As a hacker, you will often require to download new tools for hacking, but Linux automatically assigns all files and directories default permissions of 666 and 777, respectively. This means that, by default, you will not be able to execute a file immediately after downloading it. If you try, you'll usually get a message that says something like "Permission denied." For these cases, you will need to give yourself root and execute permissions using chmod to run the file.

Special Permissions

Besides the three permissions that are general-purpose, rwx, Linux has three special permissions that are slightly more complicated. They are the sticky bit, set group ID (or SGID), and set user ID (or SUID). Below is a brief explanation of each of them.

a) Granting Temporary Root Permissions with SUID

As you should know by now, a user can execute a file only if they have permission to run that particular file. If the user only has read and write permissions, they cannot run. This may seem straightforward, but there are exceptions to this rule. You may have encountered a case in which a file requires the permissions of the root user during execution for all users, even those who are not the root. For example, a file that allows users to change their password would need access to the /etc/shadow file—the file that holds the users' passwords in Linux—which requires root user privileges to execute. In such a case, you can temporarily grant the owner's privileges to run the file by setting the SUID bit on the program. It, therefore, implies that the SUID bit says that any user can execute the file with the permissions of the owner, but those permissions don't extend beyond the use of that file. To set the SUID bit, enter a 4 before the regular permissions, so a file with a new resulting permission of 644 is represented as 4644 when the SUID bit is set Setting the SUID on a file is not something a typical user would do, but if you want to do so, you'll use the chmod command, as in chmod 4644 filename.

b) Granting the Root User's Group Permissions SGID

SGID also gives temporary elevated permissions, but it gives the permissions of the file owner's group, rather than of the file's owner. This means that, with an SGID bit set, someone without the execute permission can execute a file if the owner belongs to the group that has permission to run that file. The SGID bit works slightly differently when applied to a directory: when the bit is set on a directory, ownership of new files created in that directory goes to the directory creator's group, rather than the file creator's group. This is very useful when a directory is shared by multiple users. All users in that group can execute the file(s), not just a single user. The SGID bit is represented as 2 before the regular permissions, so a new file with the resulting permissions 644 would be represented as 2644 when the SGID bit is set. Again, you would use the chmod command for this—for example, chmod 2644 filename.

c) The Outmoded Sticky Bit

The sticky bit is a permission bit that you can set on a directory to allow a user to delete or rename files within that directory. However, the sticky bit is a legacy of older Unix systems, and modern systems (like Linux) ignore it. As such, I will not discuss it further here, but you should be familiar with the term because you might hear it in the Linux world.

Managing Processes

Hackers often need to multiprocess, and an operating system like Kali is ideal for this. The hacker may have a port scanner running while running a vulnerability scanner and an exploit simultaneously. This requires that the hacker manage these processes efficiently to best use system resources and complete the task. In this section, I'll show you how to manage multiple processes.

- **Changing Process Priority with nice**

You don't often hear the word nice used in the context of hackers, but here you will. The nice command is used to influence the priority of a process to the kernel. As you saw, when we ran the ps command, numerous processes run on the system at once, and all of them are contending for the

available resources. The kernel will have final say over the priority of a process, but you can use nice to suggest that a process should be elevated in priority. The idea behind the use of the term nice is that, when you use it, you're determining how "nice" you'll be to other users: if your process is using most of the system resources, you are not being very nice. The values for a nice range from –20 to +19, with zero being the default value. A high nice value translates to a low priority, and a low nice value translates to a high priority (when you do not want to be so friendly to other users and processes). When a process is started, it inherits the nice value of its parent process. The owner of the process can lower the priority of the process but cannot increase its priority. Of course, the superuser or root user can arbitrarily set the nice value to whatever they please.

- **Setting the Priority When Starting a Process**

For demonstration purposes, let us assume we have a process named slowprocess that is located at /bin/slowprocess. If we wanted it to speed up its completion, we could start the process with the nice command:

kali >nice -n -10 /bin/slowprocess

This command would increment the nice value by -10, increasing its priority and
allocating it more resources. On the other hand, if we want to be nice to our fellow users and processes and give slowprocess a lower priority, we could increment its nice value positively by 10:

kali >nice -n 10 /bin/slowprocess

- **Changing the Priority of a Running Process with renice**

The renice command takes absolute values between –20 and 19 and sets the priority to that particular level, rather than increasing or decreasing from the level at which it started. Also, renice requires the PID of the process you are targeting rather than the name. So, if slowprocess is using an excessive amount of resources on your system and you want to give it a lower priority, thus allowing other processes a higher priority and more resources, you could renice the slowprocess (which has a PID of 6996) and give it a much higher nice value, like so:

kali >renice 20 6996

As with nice, only the root user can renice a process to a negative value to give it a higher priority, but any user can be nice and reduce priority with renice.

- **Killing Processes**

At times, a process will consume way too many system resources, exhibit unusual behavior, or, at worst, freeze. A process that displays this type of behavior is often referred to as a zombie process. For you, probably the most problematic symptom will be wasted resources used by the zombie that could be better allocated to other useful processes. When you identify a problematic process, you may want to stop it with the kill command. There are many different ways to kill a program, and each has its own kill number. The kill command has 64 different kill signals, and each does something slightly different. Here, we focus on a few you will likely find most useful. The syntax for the kill command is kill-signal PID, where the signal switch is optional. If you do not provide a signal flag, it defaults to SIGTERM.

- **Running Processes in the Background**

In Linux, whether you're working from the command line or the GUI, you're working within a shell. All commands that run are executed from within that shell, even if they run from the graphical interface. When you execute a command, the shell waits until the command is completed before offering another command prompt. At times, you may want a process to run in the background, rather than having to wait for it to complete in that terminal. For instance, say we want to work on a script in a text editor and so have called our text editor (Leafpad for instance) by entering the following:

kali >leafpad newscript

In this case, the bash shell will open the Leafpad text editor to create newscript. While we work in the text editor, the terminal is occupied with running the text editor. If we return to the terminal, we should see that it is

running our text editor and that we have no new prompt to allow us to enter more commands.

- **Moving a Process to the Foreground**

If you want to move a process running in the background to the foreground, you can use the fg (foreground) command. The fg command requires the PID of the process you want to return to the front, as shown next.

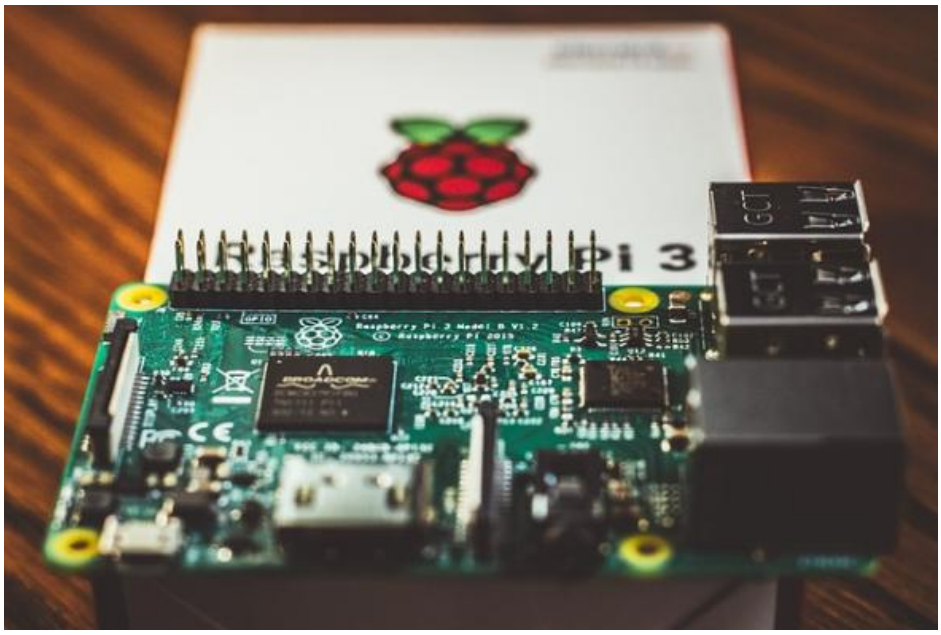
kali >fg 1234

If you do not know what the PID is, you can utilize ps to find it.

- **Scheduling Processes**

Both Linux system administrators and hackers often need to schedule processes to run at a particular time of day. A system administrator might want to schedule a system backup to run every Sunday at 1 AM, for instance. A hacker might want to set a script to run to perform reconnaissance on a specified basis, finding open ports or vulnerabilities. In Linux, you can accomplish this in at least two ways: with crond and at. The at command is a daemon, that is, a background process, useful for scheduling a job to run once at some point in the future. The crond is more suited for programming tasks to occur at intervals such as every day, week, or month.

OpenSSH and the Raspberry Pi Spy



SSH is an acronym for Secure Shell and is basically what enables us to connect securely to a terminal on a remote system, a replacement for the insecure telnet that was so common years ago. When we're building a web server, SSH enables us to create an access list (a list of users who can use this service), authenticate users with encrypted passwords, and encrypt all communication. This reduces the chance of unwanted users using the remote terminal (due to the added authentication process) or intercepting our communication (due to encryption). Probably the most widely used Linux SSH service is OpenSSH, which is installed on nearly every Linux distribution, including Kali. System administrators often use SSH to manage remote systems, and hackers often use SSH to connect to compromised remote systems, so we'll do the same here. In this example, we use SSH to set up a remote Raspberry Pi system for spying, something I call the "Raspberry Spy Pi." For this, you'll need a Raspberry Pi and the attendant
Raspberry Pi camera module. Before we do that, though, start OpenSSH on your Kali system with the now-familiar command:

kali >service ssh start

We shall be using SSH to build and control a remote spying Raspberry Pi. If you are not already familiar with it, the Raspberry Pi is a tiny but powerful, credit card-sized computer that works great as a remote spying

tool. We will employ a Raspberry Pi with a camera module to use as a remote spying device. You can purchase a Raspberry Pi at nearly any electronics retailer, including Amazon. Here, we are going to use the Raspberry Spy Pi on the same network as our Kali system, which allows us to use private, internal IP addresses. Of course, when hacking in the real world, you would probably want to set it up on another remote network, but that would be beyond the scope of this book.

- **Setting up the Raspberry Pi**

Make sure that your Raspberry Pi is running the Raspbian operating system; this is simply another Linux distribution specially ported for the Raspberry Pi CPU. You can find download and installation instructions for Raspbian at https://www.raspberrypi.org/downloads/raspbian/. Nearly everything you have learned in this book applies to the Raspbian OS on the Raspberry Pi as well as Kali, Ubuntu, and other Linux distributions. Once you have your Raspbian OS downloaded and installed, you'll need to connect your Raspberry Pi to a monitor, mouse, and keyboard and then connect it to the internet. If this is all new to you, check out the instructions at https://www.raspberrypi.org/learning/hardwareguide/ . With everything set up, log in with the username pi and the password raspberry.

Building the Raspberry Spy Pi

The first step is to make sure that SSH is running and enabled on the Raspberry Spy Pi. SSH is usually off by default, so to allow it to, go to the Preferences menu and launch the Raspberry Pi Configuration. Then go to the Interfaces tab and, next to SSH, click Enabled (if it is not already checked) and click OK. When SSH is enabled, you can start it on your Raspberry Spy Pi by opening a terminal and entering the following:

```
kali >service ssh start
```

Next, you need to attach your camera module. If you are using a Raspberry Pi version 3 board, there's only one place to connect it. Switch the Pi off, attach the module to the camera port, and then switch it on again. Note that the camera is very fragile and must never come into contact with the general-purpose Input/output pins; otherwise, it might short and die. Now,

with the SSH service up and running, place the Raspberry Spy Pi somewhere within your home, school, or some other location you want to spy on. It must, of course,
be connected to the local area network, either by Ethernet cable or, ideally, via WiFi. Now, you need to obtain the IP address of your Raspberry Pi. As previously learned, you can get a Linux device's IP address by using ifconfig:

```
pi >ifconfig
```

We are going to use the IP address 192.168.50.5 for Pi in this book. Therefore, ensure you are using the IP address of your Raspberry Spy Pi wherever you see this address appearing in this chapter. Now, from your Kali system, you should be able to connect directly to and control your Raspberry Spy Pi and use it as a remote spying system. In this simple example, your system will need to be on the same network as the Pi. To connect to the remote Raspberry Spy Pi via SSH from your Kali system, enter the following, remembering to use your own Pi's IP address.

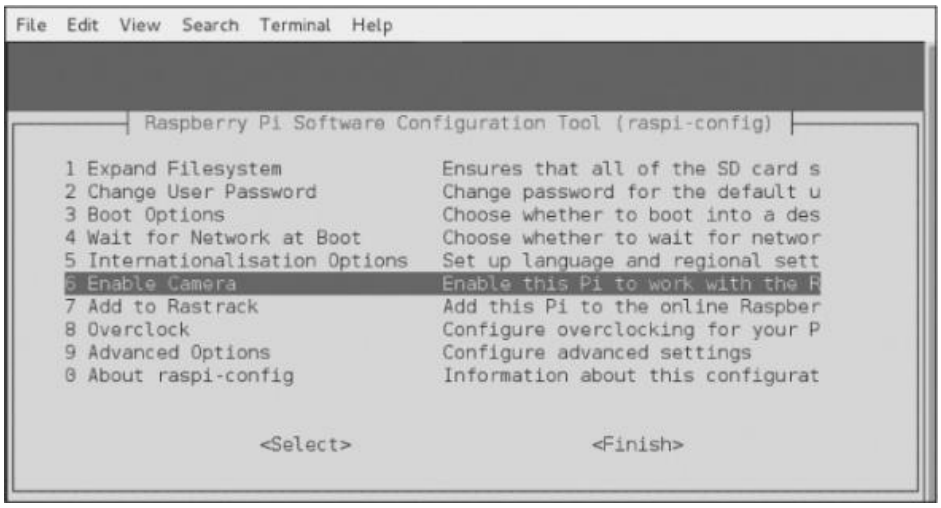
```
kali >ssh pi@192.168.50.5
pi@192.168.50.5 's password:
```

Configuring the Camera

Next, we need to configure the camera. To do so, start the Raspberry Pi configuration tool by entering the following command:

```
pi >sudo raspi-config
```

This should start a graphical menu like the one shown below:



Scroll down to 6 Enable Camera and press ENTER. Now, scroll to the bottom of this menu and select Finish and press ENTER.

Information Extraction from MYSQL

The database that is most widely used in database-driven web applications is MySQL. No doubt about it. In the modern era, where nearly every website is database-driven, this means MySQL holds the data for most of the web.
Like Linux, MySQL is open source and general public licensed (GPL), and you'll find it preinstalled on nearly every Linux distribution. Being free, open-source, and powerful, MySQL is the preferred choice for many web applications. Examples are popular websites like YouTube, WordPress, Facebook, and so on.

Start MySQL

As you would guess, Kali Linux comes with MySQL pre-installed. To start your MySQL service, enter the following into the terminal:

kali >service mysql start

Next, you need to authenticate yourself by logging in.

Setting MySQL Password

Let's see what users are already in our MySQL system by entering the following. (Note that commands in MySQL are terminated with a

semicolon.)

mysql >select user, host, password from mysql.user;

Let us assign a password to root. To do so, we shall first select a database to work with. MySQL on your system will come with some databases already set up. Use the show databases; command to see all the available databases:

mysql >show databases;

MySQL comes with three databases by default, two of which information_schema and performance_schema) are administrative databases that we will not use here. We will use the nonadministrative database, mysql, which has been included for your purposes. To begin using the mysql database, enter:

mysql >use mysql

This command connects us to mysql. Now, we can set the password for the root user to hackers-arise with the following command:

mysql >update user set password = PASSWORD("hacking-tutorial") where user = 'root';

This command will update the user by setting the user's root password to hacking-tutorial.

Accessing a Remote Database

To access a MySQL database on the localhost, we use the following syntax:

kali >mysql -u <username> -p

To access a remote database, then, we are required to give the IP address or the hostname of the system which is being used to host the MySQL database. See the example below:

kali >mysql -u root -p 192.168.10.1

The command above will link us to the MySQL database instance at the IP address 192.168.10.1 and will ask us for a password.

Connecting to a Database

With access to the system, we want to snoop around. The next logical step will be to ascertain if there exist any databases that are worth accessing. To do this, you will need to discover the databases located on the system you have had access to:

mysql >show databases;

Just like the norm in other database management systems, you will be able to connect to the database you want just by entering use databasename; in MySQL.

Examining Data

For you to view data located on the table, you will need to make use of the SELECT command. For this command to work, it needs the information below:

- The table that holds the data you want to view
- The columns within that table that contain the data you wish to view

We lay this out in the following format:
SELECT columns FROM table

Chapter 7: Cyber Security

Introduction

We define cybersecurity as being the protection of computer systems, computer networks, and their associated programs from attacks that are of a digital form. Typically, cyberattacks are carried out with the intent of gaining access, modification, or even destruction of information that is sensitive. They also attempt to extorting money from victims and are meant to interrupt the normal processes of a business.

Confidentiality, Integrity, and Availability

The three are famously referred to as the CIA triad. We can describe it as a model whose purpose is to guide information security policies within any given organization. To prevent confusing the triad with the American Central Intelligence Agency, we sometimes refer to it as the AIC triad. The three elements are the most critical components of security. In our case, we can say that availability is defined as a guarantee of access that is reliable to information by people with authorization, confidentiality is said to be a set of protocols that are used to limiting access to information, and integrity is the undertaking given to show that the information at hand is both accurate and trustworthy.

a) **Confidentiality:**

This is a rough equivalent of privacy. While ensuring that the right people can have access to crucial information, it is also prudent that vigorous measures are undertaken to make sure that there is confidentiality. There should be restricted access to the data in question by those who are authorized to view it. Out there, it is not uncommon to categorized data based on the type and amount of damage that can result from it falling into unauthorized persons. Stringent measures can more or less be implemented depending on these categories. Guarding the confidentiality of data sometimes requires specialized training for authorized to view/use persons. It would generally involve security risks that could harm that information. It can, without a doubt, help people with the proper authorization to get to know the various risk factors and equip them with countermeasures. Additional aspects of the training may comprise best practices in password-related issues alongside social engineering mechanisms.

This will help them avoid breaching rules governing data-handling with potentially disastrous results in as much as they may have intentions we can describe as being noble. For example, using a routing number or an account number is an effective measure that can be used to ensure confidentiality. We can also employ the use of data encryption to make sure that there is confidentiality. Passwords and user IDs are part of a standard procedure that is becoming a common phenomenon, two-factor authentication. There are different options. They include security tokens (soft tokens or key fobs) and biometric verification.

Furthermore, it is incumbent upon the users to take precautions in ensuring that locations where their information appears and the number of times required to send it to complete a transaction is at a minimal. In cases where we have critical data, extra measures may be necessary. Such actions can involve storing the information on disconnected storage devices, on air-gapped computers, or it can even be stored in the form of hard copies only.

a) **Integrity:**

This component of the triad comprises ensuring the trustworthiness, consistency, and accuracy of data throughout its complete life cycle. It is of immense importance that data that is in transit is not altered. Solid steps need to be taken to make sure that no modification on the data by unauthorized people happens. For instance, in cases where we have a confidentiality breach. Here, the countermeasures can involve user access controls and file permissions. To prevent accidental deletion or erroneous changes by authorized users, we can employ the use of version control. In place, there also need to exist mechanisms to help in the detection of data changes which may result from non-human events, including a server crash or an electromagnetic pulse. We can include checksums and cryptographic checksums to help with the integrity verification of data. Lastly, it may be necessary to have some form of redundancies and backups that will help in the restoration back to its former state.

b) Availability:

The rigorous maintenance of all the hardware ensures that there will always be an availability for the services rendered by this hardware. Failing equipment should be promptly and adequately repaired to keep in order a properly functioning operating system environment that is devoid of any software conflicts. One aspect of maintenance that should also be carried out is updating all the necessary system components. It will also be to provide ample bandwidth for communications and to ensure a minimal occurrence of bottlenecks. Mitigation of hardware failures and their repercussions can be done using high-availability clusters, redundancy, RAID, and even failovers.

For the worst-case scenarios that occur, disaster recovery that is both adaptive and fast is essential. For this to be possible, the disaster recovery plan laid down has to be comprehensive. Prevention of data loss or connection interruptions needs to also account for unpredictable events. Examples include fire and natural disasters. Copies of back up data can be securely stored at a location that is geographically-isolated to prevent loss of data resulting from such occurrences. Such sites also need to be water and fire-resistant. To guard against issues such as downtime and inaccessibility of data due to denial-of-service attacks and network

intrusions, we can employ the use of extra security equipment, for instance, proxy servers, firewalls, and software.

Issues Arising from the CIA:

The CIA paradigm faces immense challenges where big data is involved. This is primarily because of the sheer volume needing to be kept safe, the variety of formats of the data and finally the multiplicity of the originating sources. Disaster recovery plans and duplicate sets of data all make the already high cost even higher. Additionally, oversight is often lacking since the main objective of big data is for analytics purposes, i.e., gathering data and using it to make some kind of useful interpretation. We all know this fellow, Edward Snowden who brought this issue to light. Security agencies carry out the collection of enormous volumes of peoples' private data throughout the world. To safeguard individual information from exposure in the IoT environment, we have special considerations known as the Internet of Things privacy. This means that almost any logical or physical entity can be assigned a unique identifier to enable autonomous communications over a network, including the Internet.

The transmitted data from a particular endpoint may not, on its own, necessarily result in any privacy issues. The catch is, however, when the fragmented data from multiple endpoints is accessed, gathered, and analyzed, sensitive information can be obtained. Securing the Internet of Things is itself a formidable challenge since it comprises numerous Internet-enabled devices besides computers. Such devices are in most cases, often set up with default passwords that are weak or in some cases, the devices are unpatched. Unless IoT is protected adequately, there is a likelihood that it may be used as a separate vector of attack or be made a part of a thingbot. Recently, it has been demonstrated by researchers that it is possible to compromise a network just by using a Wi-Fi-enabled light bulb. It is essential for us that we consider the security of the numerous network-capable products that are under development.

Encryption

We define encryption as a mechanism through which plaintext or other data type are changed from their currently readable form to an encoded way. It is only an entity having access to a decryption key that can decode the data. This is an important measure that usually is used to provide end-to-end data security across networks. Encryption, as a proactive security measure, is commonly used all over the internet for purposes of protecting crucial information belonging to users which is being exchanged between servers and browsers. That can include private information such as payment information, passwords, and other personal information. Individuals, together with organizations, may also opt to use encryption to ensure the safety of sensitive data that is stored on mobile devices, servers, and computers.

a) How Encryption Works

Plaintext data, also known as unencrypted data, is encrypted through the use of an encryption algorithm plus an encryption key. The result of this is a ciphertext that can be seen only in its original form if decrypted with the correct key. On the other hand, decryption is the reverse of encryption. The steps used in encryption are followed in a reverse fashion. In the modern age, we have two commonly used encryption algorithms. They are symmetric and asymmetric encryptions.

When it comes to the symmetric encryption mechanism, a single key is utilized for encryption. The Advanced Encryption Standard (AES) is one of the most used symmetric-key ciphers. It was designed primarily to protect classified information for governments. This mechanism is faster in comparison to asymmetric encryption. The sender must, however, share the encryption key with the recipient. The keys need to be managed in a secure fashion. This uses an asymmetric algorithm in most cases.

On the other hand, we have asymmetric cryptography. We can also refer to it as public-key cryptography. Here, two different keys are used. They are, however, mathematically linked. The keys are as follows; one key is public and the other one private. The public key many times can be shared with anyone. The private key has to be kept secret. In asymmetric cryptography, the commonly used encryption algorithm is the RSA. The reason is to some extent that the two keys can encrypt a message, which is to imply the key that is opposite to the one used for the encryption is used to decrypt it. This feature offers a way of ensuring that we not only have confidentiality but also authenticity, non-reputability, and integrity of electronic communications and data.

b) Benefits of Encryption

Confidentiality of digital data which is stored on computer systems or that which is sent through the internet or any other computer network is protected by using encryption. Organizations such as Payment Card Industry Data Security Standard (PCI DSS) require that sensitive data be encrypted to keep unauthorized entities from accessing the data. We also have some standards requiring or recommending data encryption. Nowadays, modern encryption algorithms serve an integral role in making sure that the security of communications and IT systems possess not only confidentiality but also the under listed key elements of security:

- Authentication: the origin of a given message should be able to be verified.
- Integrity: This has got to do with keeping the message intact. That is, the contents of messages have not been altered or deleted from the time it was sent.

- Nonrepudiation: Here, non-repudiation means that a particular sender cannot dispute that they send the message.

Backup and Redundancy

Usually, we use backup where copies of data are created in anticipation of a catastrophic loss. On the other hand, redundancy is a lot more than just data storage. Redundancy aims to provide a continuity of service regardless of what will happen. Data redundancy ensures that the storage of data is done at multiple and heterogeneous locations. We also have what we call network redundancy whereby a given network is configured in such a way that it has numerous alternative systems. The alternative systems serve to ensure continuity of service regardless of what happens.

Data Redundancy

For any organization, it is essential first that regular services are restored as soon as possible after there has been a security breach. Data should be able to be reconstructed as quickly as possible. To this end, businesses have come up with various ways to make sure there is data redundancy. It is common knowledge that these methods come with their own merits in terms of cost-effectiveness, speed, and management. The most common way is using off-site tape backups. In this method, magnetic tapes are used to store a complete bit-for-bit copy of a storage volume. The tapes can be transferred to an off-site storage facility where they can be easily retrieved whenever there is a catastrophic failure. Besides, we can use Cloud Backup to safeguard data against losses.

Network Redundancy

Most of the infrastructure we use for our networks are unbelievably fragile. For instance, when a router burns out due to one reason or another, the result is that there will be a prolonged period of network downtime. To mitigate against this, businesses make sure that networks they use have an adequate redundancy so that they can survive and provide services in cases of an emergency. Fundamentally, network redundancy means that no matter what type of failure occurs, a network will still be up and running. To be able to do this, we can have multiple network devices such as hubs, routers, and switches configured to stand in for one of them that fails. We

also have ISP redundancy where a gateway in the network is joined to more than one separate ISP. Just like with the devices, one ISP will take over whenever there is a failure. In cases where a network is functioning correctly, we can use the ISPs to share the traffic resulting in reduced congestion of the network. This here is called load sharing.

Preventing a SPOFF

SPOFF is full for a single point of failure. We do not desire that one critical part of a system failure can render the entire system unusable. Any planning needs to mitigate this phenomenon. A single point of failure can be reduced or eliminated by way of redundancy. This will make sure that there is not a single component that can prevent the proper working of a system.

Chapter 8: Becoming Secure and Anonymous

Introduction

Today, nearly everything we do on the internet is tracked. Whoever is doing the tracking, whether it be Google tracking our online searches, website visits, and email or the National Security Agency (NSA) cataloging all our activities, all our online moves are being recorded, indexed, and then mined for someone's benefit. The average individual and the hacker, in particular, need to understand how to limit this tracking and remain relatively anonymous on the web to limit this ubiquitous surveillance. In this chapter, we look at how you can navigate the World Wide Web anonymously (or as close as you can get) using four methods:

- The Onion Network
- Proxy servers
- Virtual private networks
- Private encrypted email

No one method is sure to keep your activities safe from prying eyes, and given enough time and resources; anything can be tracked. However, these methods will likely make the tracker's job much more difficult.

How the Internet Gives Us Away

To begin, let's discuss at a high level some of the ways our activities on the internet are tracked. We won't go into all tracking methods, or into too much detail about any one process, as that would be beyond the scope of this book. Indeed, such a discussion could take up an entire book on its own. First, your IP address identifies you as you traverse the internet. Data sent from your machine is generally tagged with your IP address, making your activities easy to track.
Second, Google and other email services will "read" your email, looking for keywords to more efficiently serve your ads. Although there are many more sophisticated methods that are far more time and resource-intensive, these are the ones we try to prevent in this chapter. Let's start by taking a look at how IP addresses give us away on the internet.

When you send a packet of data across the internet, it contains the IP addresses of the source and destination for the data. In this way, the packet knows where it is going and where to return the response. Each packet hops through multiple internet routers until it finds its destination and then jumps back to the sender. For general internet surfing, each hop is a router the packet passes through to get to its destination. There can be as many as 20–30 hops between the sender and the destination, but usually, any packet will find its way to the destination in fewer than 15 hops.

As the packet traverses the internet, anyone intercepting the packet can see who sent it, where it has been, and where it is going. This is one-way websites that can tell who you are when you arrive and log you in automatically, and it is also how someone can track where you've been on the internet. To see what hops a packet might make between you and the destination, you can use the traceroute command, as shown next. Enter traceroute and the destination IP address or domain, and the command will send out packets to the destination and trace the route of those packets.

The Onion Router System

In the 1990s, the US Office of Naval Research (ONR) set out to develop a method for anonymously navigating the internet for espionage purposes. The plan was to set up a network of routers that was separate from the internet's routers, that could encrypt the traffic, and that only stored the unencrypted IP address of the previous router— meaning all other router

addresses along the way were encrypted. The idea was that anyone watching the traffic could not determine the origin or destination of the data. This research became known as “The Onion Router (Tor) Project” in 2002, and it’s now available to anyone to use for relatively safe and anonymous navigation on the web.

How Tor Works

Packets sent over Tor are generally not sent over the regular routers so closely monitored by so many but instead are sent over a network of over 7,000 routers around the world, thanks to volunteers who allow their computers to be utilized by Tor. On top of using an entirely separate router network, Tor encrypts the data, destination, and sender IP address of each packet. At each hop, the information is encrypted and then decrypted by the next hop when it’s received. In this way, each packet contains information about only the previous hop along the path and not the IP address of the origin. If someone intercepts the traffic, they can see only the IP address of the previous hop, and the website owner can see only the IP address of the last router that sent the traffic.

This ensures relative anonymity across the internet. To enable the use of Tor, you need to install the Tor browser from https://www.torproject.org/. Once installed, you can use it like any old internet browser. By using this browser, you will be navigating the internet through a separate set of routers and will be able to visit sites without being tracked by Big Brother. Unfortunately, the tradeoff is that surfing via the Tor browser can be a lot slower; because there are not nearly as many routers, the bandwidth is limited in this network.

In addition to being capable of accessing nearly any website on the traditional internet, the Tor browser is capable of accessing the dark web. The sites that make up the dark web require anonymity, so they allow access only through the Tor browser, and they have addresses ending in .onion for their top level
domain (TLD). The dark web is infamous for illegal activity, but there exist quite a number of legal services that are also available there. A word of caution, however: when accessing the dark web, you may come across material that many will find offensive.

a) Security Concerns

The intelligence and spy services of the United States and other nations consider the Tor network a threat to national security, believing such an anonymous network enables foreign governments and terrorists to communicate without being watched. As a result, we have numerous robust and ambitious research projects are working to break the anonymity of Tor. Tor's anonymity has been broken before by these authorities and will likely be broken again. The NSA, as one instance, runs its own Tor routers, meaning that your traffic may be traversing the NSA's routers when you use Tor.

If your traffic is exiting the NSA's routers, that's even worse, because the exit router always knows your destination. The NSA also has a method known as traffic correlation, which involves looking for patterns in incoming and outgoing traffic, that has been able to break Tor's anonymity. Though these attempts to break Tor won't affect Tor's effectiveness at obscuring your identity from commercial services, such as Google, they may limit the browser's effectiveness in keeping you anonymous from spy agencies.

Proxy Servers

Another strategy for achieving anonymity on the internet is to use proxies, which are intermediate systems that act as middlemen for traffic: the user

connects to a proxy, and the traffic is given the IP address of the proxy before it's passed on.
When the traffic returns from the destination, the proxy sends the traffic back to the source. In this way, traffic appears to come from the proxy and not the originating IP address. Most probably, the proxy will keep a log of your traffic. However, an investigating entity would need to obtain a search warrant or subpoena in order for them to obtain the logs. To make your traffic even harder to trace, you can use more than one proxy, in a strategy known as a proxy chain, which we'll look at a little later in this chapter. Kali Linux has an excellent proxying tool called proxychains that you can set up to obscure your traffic. The syntax for the proxychains command is straightforward, as shown here:

kali >proxychains <the command you want proxied> <arguments>

The arguments you provide might include an IP address.

Setting Proxies in the Config File

In this section, we set a proxy for the proxychains command to use. As with nearly every application in Linux/Unix, the configuration of proxychains is managed by the config file—specifically /etc/proxychains.conf. Open the config file in your text editor of choice with the following command (replacing Leafpad with your chosen editor if necessary):

kali >leafpad /etc/proxychains.conf

You should see a proxychains.conf file. Scroll down this file to line 61, and you should see the ProxyList section. We can add proxies by entering the IP addresses and ports of the proxies we want to use in this list.

a) Security Concerns

As a last note on proxy security, be sure to choose your proxies wisely: proxychains is only as good as the proxies you use. If you are intent on remaining anonymous, do not use a free proxy, as mentioned earlier. Hackers use paid for
proxies that can be trusted. The free proxies are likely selling your IP address and browsing history. As Bruce Schneier, the famous

cryptographer and security expert, once said, “If something is free, you’re not the customer; you’re the product.” In other words, any free product is likely gathering your data and selling it. Why else would they offer a proxy for free? Although the IP address of your traffic leaving the proxy will be anonymous, there are other ways for surveillance agencies to identify you. For instance, the owner of the proxy will know your identity and, if pressured enough by espionage or law enforcement agencies with jurisdiction, may offer up your identity to protect their business. It is good to be aware of the limitations of proxies as a source of anonymity.

Virtual Private Networks

Using a virtual private network can be an effective way to keep your web traffic relatively anonymous and secure. A VPN is used to connect to an intermediary internet device such as a router that sends your traffic to its ultimate destination tagged with the IP address of the router. Using a VPN can certainly enhance your security and privacy, but it is not a guarantee of anonymity. The internet device you connect to must record or log your IP address to be able to send the data back to you accurately, so anyone able to access these records can uncover information about you.

The beauty of VPNs is that they are simple and easy to work with. You can open an account with a VPN provider and then seamlessly connect to the VPN each time you log on to your computer. You would use your browser as usual to navigate the web, but it will appear to anyone watching that your traffic is coming from the IP address and location of the internet VPN device and not your own. Besides, all traffic between you and the VPN device is encrypted, so even your internet service provider cannot see your traffic.

Among other things, a VPN can be useful in evading government-controlled Content and information censors. For instance, if your national government limits your access to websites with particular political messages, you can likely use a VPN based outside your country to access that Content. Some media corporations, such as Netflix, limit access to their Content to IP addresses originating from their nation. Using a VPN based in a country that those services allow can often get you around those

access limitations. Some of the best VPN services are: IPVanish, NordVPN, ExpressVPN, CyberGhost, Golden Frog VPN, Hide My Ass, Private Internet Access, PureVPN, TorGuard, and Buffered VPN

The strength of a VPN is that all your traffic is encrypted when it leaves your computer, thus protecting you against snooping, and your IP address is cloaked by the VPN IP address when you visit a site. As with a proxy server, the owner of the VPN has your originating IP address.

IPsec

IPsec is used to provide data integrity, authentication, and confidentiality between two points across the IP network that are in communication. It is an Internet Engineering Task Force protocol that also provides definitions of the encrypted, decrypted and authenticated packets. Additionally, key management and secure key exchange protocols are defined in IPsec.

a) Functions of IP Security

The following are tasks that can be done by IPsec:

- For encryption of data found in the application layer.
- Securing routers transmitting data over the internet.
- IPsec provides us with authentication without there being any encryption.
- It does the safeguarding of data on the network through the creation of circuits using IPsec tunneling. This works just like the Virtual Private Network.

b) IP Security Components

Below are some of the components that comprise an IPsec:

i. **Internet Key Exchange (IKE)** – This is a protocol for network security that has been designed to exchange encryption keys dynamically as well as bypass the Security Association (SA) between two devices. SA is used for the establishment of security attributes that are shared between any two network elements. These attributes are what support secure

communications. Additionally, IKE offers protection to contents of messages plus an open frame that can be used for the implementation of standard algorithms, for instance, MD5 and SHA.

ii. **Encapsulating Security Payload (ESP)** – This is used to ensure data integrity, authentication, anti-replay, and encryption. ESP also does payload authentication.

iii. **Authentication Header (AH)** – IPsec uses an authentication header to ensure there is data integrity, anti-replay, and authentication. An authentication header, however, does not offer encryption. The anti-replay protection function is used to guard against unauthorized packet transmission, but it does not keep the data confidential.

c) The Operation of IP Security

1. First, a host will check to see if a packet needs to be send using IPsec or not. It is the traffic of the packets that trigger the security policy on their own.

2. Phase 1 of the internet key exchange begins with the two hosts that are utilizing IPsec authenticating themselves to each other. That will start a secure channel. Here, we have two modes:

- The Main mode which is used for the provision of the greater security and;
- The Aggressive mode that makes a host be able to create an IPsec circuit expeditiously.

3. Using the above channel (established in step 2), negotiation on the manner in which the IP circuit will encrypt data across the IP circuit will be done.

4. After that, the Phase 2 internet key exchange happens over the secure channel that was negotiated.

5. Data exchanged is then carried out over the newly established IPsec encrypted tunnel. Encryption and decryption of packets is carried out by the hosts through IPsec SAs.

6. Upon the completion of communication, or time-out of a session between the hosts, the IPsec tunnel will be terminated. Both the hosts will discard the keys.

Chapter 9: Cryptography

Introduction

As hackers, we are often faced with the hurdle of cryptography and encryption. In some cases, we use it to hide our actions and messages. Many applications and protocols use encryption to maintain the confidentiality and integrity of data. To be able to crack passwords and encrypted protocols such as SSL and wireless, you need to be at least familiar with the concepts and terminology of cryptography and encryption. To many new hackers, all the ideas and terminology of cryptography can be a bit overwhelming and opaque. With this brief overview for the newcomer, I hope to lift the fog that shrouds this subject and shed a tiny bit of light on cryptography.

There are so much mathematics and algorithms in encryption, and that is a topic we would not rather venture into at this point. The explanations will be quite simple and surprisingly easy to understand. We are going to look at the basic concepts and terminologies so that you will be in a position to know some related topics whenever they come up. These include wireless cracking, password cracking, encryption technologies, and hashing. My intention, however, is not to make a cryptographer out of you here. It is a skill that requires time to hone, but to help familiarize the beginner with

the terms and concepts of cryptography to help you become a credible hacker.

A Word About Key Size

Key size matters a lot when it comes to cryptography. More secure encryptions have larger keys. A 256-bit key AES is therefore much stronger as compared to a 128-bit key AES. That means it is also much difficult to break it. It suffices to say that in encryption that employs the use of a similar algorithm, the larger the size of the key, the stronger the encryption will be. However, note that the encryptions' strength is based on the key size and the specifics of the algorithm as well. This, therefore, does not imply that larger keys denote stronger encryption between the various encryption algorithms. Let's get started by breaking encryption into categories.

a) Types of Cryptography

Below are the kinds of encryptions we are going to concentrate on in this book.

- Asymmetric Encryption
- Symmetric Encryption

In this book, however, we are going to focus on symmetric and asymmetric encryption.

Symmetric Cryptography

Here, both the sender and receiver possess similar keys. Symmetric cryptography is undoubtedly the commonly used form of cryptography today. Picture this; you encrypt a message using a password. Supposing I have the same password, I will be able to access the encrypted message. Any other person will not read that message. See how easy that is! This type of cryptography is high-speed and is well suited for streaming applications or bulk storage. A major stumbling block with this method of cryptography is the key exchange. As in the example we have seen above, if we have two ends requiring similar keys, what they need is another third

channel that can be used to exchange the keys. This is where symmetric cryptography has its biggest weakness. Assume the entities intending to exchange messages are miles apart, how then, can the key be exchanged. As you may already be aware, the aspect of confidentiality arises. The entities can decide to exchange the key via email, mail, telephone, and so on. That makes it possible to intercept the key that is being exchanged, and as such, the encryptions' confidentiality will be compromised. We have many symmetric algorithms currently in use. The common ones are briefly discussed below.

1. **DES** – Developed by IBM, DES was among the pioneer encryption schemes. Later own, DES was discovered to possess flaws and was breakable as well. It was DES encryption that was used in hashing early systems of LANMAN originally (pre-2000).

2. **3DES** – It is the flaws in DES that occasioned the development of this encryption algorithm. It works by a triple application of the DES hence its name. That makes it a bit more secure when compared to DES.

3. **AES** – In full, AES stands for Advanced Encryption Standard. Cryptographically speaking, AES is not an encryption algorithm by itself. It was NIST that developed AES. It is one of the most robust encryptions in use today. AES utilizes the 128-, 196-, and 256-bit key. Since 2001, AES has been occupied by the Rijndael algorithm. This standard is commonly used in SSL/TLS, WPA2, among other protocols that need speed and confidentiality.

4. **RC4 -** This does encryption of each bit or byte instead of a single block of information. This is called streaming. RC4 was designed and created by RSAs' Rivest Ronald. This method of encryption is commonly used in WEP and VoIP applications.

5. **Blowfish** – Blowfish utilizes a key with a varying length. It is a very secure encryption scheme. It is additionally open-source,

and as such, anyone can be able to use it without a license.

6. **Twofish** – It is similar to Blowfish. It, however, possesses advanced capabilities such as the use of the 128 or 256-bit key. Twofish was, at some point, a strong contender for AES. Examples of applications using Twofish include cryptcat and OpenPGP, among others. Additionally, it is not patented, just like Twofish.

Asymmetric Cryptography

This type of cryptography utilizes different keys for the two ends of the channel of communication. It is an astonishingly slow technique that, when compared to symmetric cryptography, is about a thousand times slower! It is, therefore, an undesirable method for use where bulk encryption or streaming communication is concerned. On a positive note, it solves the problem of key exchange. This is because there is no need for having the same keys at both ends of a communication. This type of cryptography is predominantly utilized in cases where two entities need to exchange information but are unknown to each other. The information being transferred here usually is in the form of small bits, for instance, identifying information, i.e., a certificate or a key. Due to limitations in speed, asymmetric cryptography is not generally used for bulk or streaming encryption. Below are some schemes found under asymmetric encryption.

1. **Diffie-Hellman** – Without any doubt, Diffie-Hellman key exchange can be said to be the most exceptional development in cryptography. Diffie and Hellman came up with a method of key generation. This effectively eradicated the problem of key exchange that is often a characteristic of symmetric key encryption.

2. **RSA** – This is an abbreviation for Rivest, Shamir, and Adleman. This scheme makes use of a method where very large prime numbers are factorized. The result is used as the relationship between the two keys.

3. **PKI** – this is a Public key infrastructure mainly used for exchanging confidential information in an asymmetric system. PKI makes use of a public key alongside a private key.

4. **ECC** – this is short for Elliptical curve cryptography. The scheme is slowly but surely gaining popularity in the world of mobile computing. This is because it is efficient and also requires minimal energy consumption and computing power to provide a similar level of security. The scheme is dependent on the relationship that is shared by two functions that are located on the same elliptical curve.

5. **PGP** – an abbreviation for Pretty Good Privacy that makes use of encryption that asymmetric for purposes of ensuring the integrity and privacy of email messages.

Data Security

For us to minimize unauthorized access to databases, websites, and computers, we need measures that can safeguard digital privacy. These measures are what we call data security. It serves to guard data against corruption. For all organizations, big and small alike, data security is a

crucial IT aspect. Sometimes it can be referred by the name computer security or information security. Common technologies used for data security comprises of data masking, backups, and also data erasure, among many others. Encryption is also a data security technology that is essential in safeguarding the privacy of data as we have said. Here, hardware, software, hard drives, and digital data are encrypted. This is to make sure that they are unreadable to hackers and other users. Here we are talking about those who are unauthorized that may get their hands on the hardware or software.

Authentication is one way of practicing data security. It is likely that you have encountered a scenario where you needed a password to log into your device or even to access your email. Users must provide identifying credentials such as biometric data, a password, a username, and so on to do a verification of their identities before granting them access to data or a system.

Digital Certificates

A digital certificate is used in the authentication of the web credentials of a particular sender. The certificate also allows the receiving entity of an encrypted message to get to understand that the data is from a source that is trusted. A certification authority issues the digital certificate. Message encryption and self-signatures use digital certificates. Identity certificates or public key certificates is the other name we use for digital certificates. X.509 is an example of a commonly used digital certificate.

Conclusion

May I take this opportunity to thank you for being able to make it to the end of Hacking with Kali Linux, let's hope it has been edifying and through it, you have been able to accrue the requisite knowledge to enable you to begin your hacking career or improve your skills if you are already one. I sincerely hope that you have enjoyed flipping pages all the way from the first topic which was; Basics of Hacking, Cyber Attacks, Linux for Hacking, Basics of Kali, Scanning and Managing Networks, File and Directories Permissions, Cyber Security, Becoming Secure and Anonymous, and finally onto some basics of cryptography. I am also

hoping that by studying this book, you have got to learn plenty of practical concepts that you need to become a hacking expert.

By now, you must have been able to get access to a vast body of theoretical knowledge regarding the various types of attacks that can be launched on your systems, the reason for launching them, and how you can safeguard your infrastructure against such attacks. These are your first steps towards becoming a professional hacker. The book covers topical issues like wireless network attacks, cyber-attacks, and penetration testing, among others. It, therefore, means that you are now in an excellent position to discern network attack mechanisms being perpetrated in the real world and recommend appropriate remedial measures.

I have also given you several security measures you can implement to keep your networks safe. The formatting is such that the language is quite user-friendly and that you can understand the importance of securing your systems. Going forward, the next step is to put the concepts you have acquired from this book into practice. They say practice makes perfect, and it is by practicing that one can become a master in the field of hacking, more so using Kali Linux. Let the knowledge you have acquired from the book work for you.