

Chapter 2

Linux Basics

In order to become a good ethical hacker or penetration tester, you need to be conversant with Linux, which is by far one of the most powerful operating systems. Linux is really good for ethical hacking and penetration testing because it is compatible with a wide variety of related tools and software, whereas other operating systems such as Mac and Windows support fewer of these software and tools. In this chapter, I will teach you some of the very basics of operating a Linux OS. If you are already familiar with Linux basics, you can skip this chapter.

One of the most common questions asked in many forums is “Which Linux distro should I use?” As there are tons of Linux distros such as Ubuntu, Fedora, Knoppix, and BackTrack you can use any Linux distro you want as all work in a similar manner. However, I suggest you use BackTrack if you really wish to dig deeper into this subject because it is all encompassing from a penetration tester’s perspective.

Major Linux Operating Systems

Before talking about BackTrack, let’s take a look at some of the Linux-based distros that you will encounter very often:

Redhat Linux—Used mostly for administration purpose.

Debian Linux—Designed for using only in open source software.

Ubuntu Linux—Designed mostly for personal use.

Mac OS X—Used in all Apple computers.

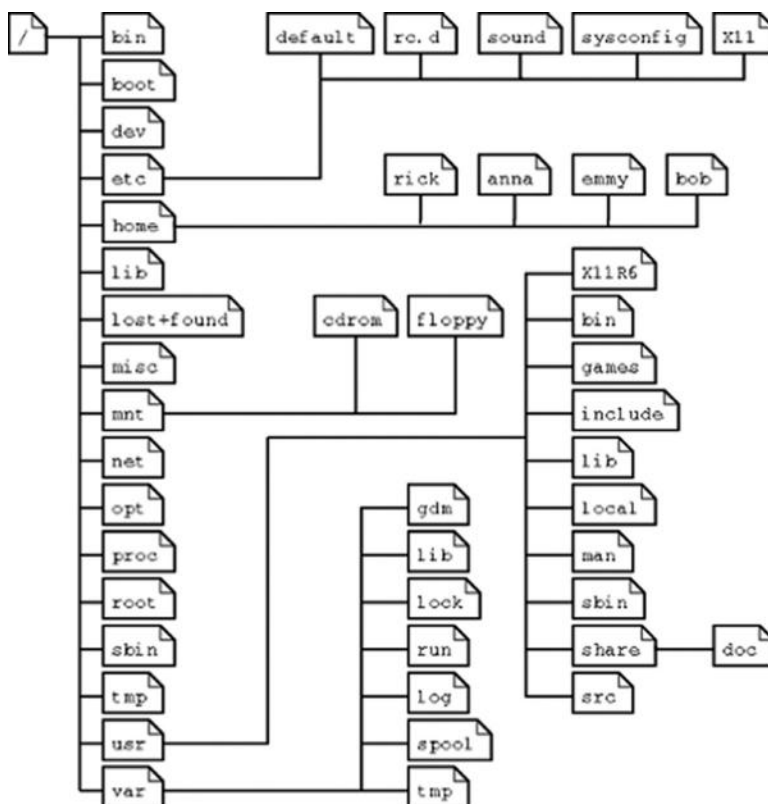
Solaris—Used in many commercial environments.

BackTrack Linux—Used mostly for penetration testing.

File Structure inside of Linux

On a Linux system, most everything is a file, and if it is not a file, then it is a process.

Here is a general diagram for file structure in Linux.



There are certain exceptions in a Linux file system

Directories—Files that are lists of other files.

Special file—The mechanism used for input and output. /dev are special files.

Links—A system to make file or directory visible in multiple parts of the systems.

Sockets—A special file type, similar to TCP/IP sockets providing inter-process networking.

Pipes—More or less like sockets; they form a way for process to communicate with each other with out using network socket.

File types in a long list:

Symbol	Meaning
-	Regular file
d	Directory
l	Link
c	Special file

s	Socket
p	Named pipe
b	Block device

Subdirectories of the root directory:

<i>Directory</i>	<i>Content</i>
/bin	Common programs, shared by the system, the system administrator, and the users.
/boot	The startup files and the kernel, vmlinuz. In some recent distributions also grub data. Grub is the GRand Unified Boot loader and is an attempt to get rid of the many different boot-loaders we know today.
/dev	Contains references to all the CPU peripheral hardware, which are represented as files with special properties.
/etc	Most important system configuration files are in/etc., this directory contains data similar to those in the Control Panel in Windows
/home	Home directories of the common users.
/initrd	(on some distributions) Information for booting. Do not remove!
/lib	Library files, includes files for all kinds of programs needed by the system and the users.
/lost+found	Every partition has a lost+found in its upper directory. Files that were saved during failures are here.
/misc	For miscellaneous purposes.
/mnt	Standard mount point for external file systems, for example, a CD-ROM or a digital camera.
/net	Standard mount point for entire remote file systems.
/opt	Typically contains extra and third-party software.
/proc	A virtual file system containing information about system resources. More information about the meaning of the files in proc is obtained by entering the command man proc in a terminal window. The file proc.txt discusses the virtual file system in detail.
/root	The administrative user's home directory. Mind the difference between /, the root directory and /root, the home directory of the root user.
/sbin	Programs for use by the system and the system administrator.
/tmp	Temporary space for use by the system, cleaned upon reboot, so don't use this for saving any work!
/usr	Programs, libraries, documentation, etc., for all user-related programs.
/var	Storage for all variable files and temporary files created by users, such as log files, the mail queue, the print spooler area, space for temporary storage of files downloaded from the Internet, or to keep an image of a CD before burning it.

File Permission in Linux

Although there are already a lot of good security features built into Linux-based systems, based upon the need for proper permissions, I will go over the ways to assign permissions and show you some examples where modification may be necessary. Wrong file permission may open a door for attackers in your system.

Group Permission

Owner—The Owner permissions apply only the owner of the file or directory; they will not impact the actions of other users.

Group—The Group permissions apply only to the group that has been assigned to the file or directory; they will not affect the actions of other users.

All User/Other—The All Users permissions apply to all other users on the system; this is the permission group that you want to watch the most.

Each file or directory has three basic permission types:

Read—The Read permission refers to a user's capability to read the contents of the file.

Write—The Write permissions refer to a user's capability to write or modify a file or directory.

Execute—The Execute permission affects a user's capability to execute a file or view the contents of a directory.

Let's see how it works.

File permission is in following format.

Owner Group Other/all

```
root@Net:~# ls -al
```

We will talk about aforementioned command later on in this chapter.

```
-rwxr-xr-x 1 net tut 77 Oct 24 11:51 auto run
drwx----- 2 ali tut 4096 Oct 25 2012 cache
```

File auto run permission

--No special permissions

rwx—Owner (net) having read, write, and execute permission while group (tut) having read and execute and other also having same permission.

File cache permission

d—Represent directory

rwx—Owner (ali) having read, write, and execute permission while group (tut) and other/all does not have any permission for accessing or reading this file.

Linux Advance/Special Permission

l—The file or directory is a symbolic link

s—This indicated the setuid/setgid permissions. Represented as a s in the read portion of the owner or group permissions.

- t—This indicates the sticky bit permissions. Represented as a t in the executable portion of the all users permissions
- i—chatter Making file unchangeable

There are two more which mostly used by devices.

- c—Character device
- b—Block device (i.e., hdd)

Let's go through some examples

Link Permission

```
root@net:~#ln -s new /root/link
root@net:~#ls -al
lrwxrwxrwx 1 ali ali 3 Mar 18 08:09 link -> new
link is created for a file name called new (link is symbolic for file name new)
```

Suid & Guid Permission

setuid (*SUID*)—This is used to grant root level access or permissions to users

When an executable is given *setuid* permissions, normal users can execute the file with root level or owner privileges. *Setuid* is commonly used to assign temporarily privileges to a user to accomplish a certain task. For example, changing a user's password would require higher privileges, and in this case, *setuid* can be used.

setgid (*SGID*)—This is similar to *setuid*, the only difference being that it's used in the context of a group, whereas *setuid* is used in the context of a user.

```
root@net:~#chmod u+s new
root@net:~#ls -al
-rwSr--r-- 1 ali ali 13 Mar 18 07:54 new
```

Capital **S** shows *Suid* for this file.

```
root@net:~#chmod g+s guid-demo
root@net:~#ls -al
-rw-r-Sr-- 1 ali ali 0 Mar 18 09:13 guid-demo
```

Capital **S** shows *Guid* for *guid-demo* file and capital **S** is in group section.

Stickybit Permission

This is another type of permission; it is mostly used on directories to prevent anyone other than the “root” or the “owner” from deleting the contents.

```
root@net:~#chmod +t new
root@net:~#ls -al
-rw-r--r-T 1 ali ali 13 Mar 18 07:54 new
```

Capital **T** shows that stickybit has been set for other user (only owner or root user can delete files)

Chatter Permission

```

root@net:~#lsattr
----- ./new
root@net:~#chattr +i new
root@net:~#lsattr
---i----- ./new

```

Small **i** shows that this file is unchangeable and **lsattr** is a command to check if there is **chattr** on file. Before we end up with file permission, let's have little look about numerical file permission.

```

r = 4
w = 2
x = 1

```

The sum of those aforementioned values manipulates the file permission accordingly, that is,

```

root@net:~# ls -al
-rw-r--r-- 1 ali ali 13 Mar 18 07:54 new

```

Here other user only having “read” permission so what we are going to do is to change it into read and write but not execute.

```

root@net:~#chmod 646 new
root@net:~#ls -al
-rw-r--rw- 1 root root 13 Mar 18 07:54 new

```

Let's explore a bit more into it, we want read + write permission so $4 + 2 = 6$ that's mean read and write. Hope it is clear now how to set permission on a file and what it does.

Most Common and Important Commands

ls:	list directory contents
cd:	changes directories
rm:	remove files or directories
chmod:	change file mode bits, from read to write and vise versa
chown:	change ownership of a file
chgrp:	change group ownership
screen:	screen manager with VT100/ANSI terminal emulation, create background process with terminal emulator.
ssh:	secure shell for remote connection
man:	manual/help
pwd:	print name of current/working directory.
cd..:	moves up one directory
mkdir:	create a new directory
rmdir:	remove director
locate:	find a file with in directory or system

Cron Permission

Two files play important role in cron.

Cron Permission

Two files play important role in cron.

```
cron.allow
cron.deny
```

If these files exist, then they impose some restriction accordingly on users. That is, if a user is in deny list, so he/she won't be able to schedule any job/task and if user is in allowed list then she/he will be able to add schedule job/task. All we have to do is just add user name in either of these two files.

Cron Files

```
Cron.daily
Cron.hourly
Cron.weekly
Cron.monthly

/etc/crontab: system-wide crontab

root@net:~#cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the 'crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts
--report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts
--report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts
--report /etc/cron.monthly )
```

This is the output for crontab file; in other words, cron.hourly , cron.daily , cron.weekly , cron.monthly are symlink of crontab.

Let's say I would like to run a schedule at 12Am daily basis .

```
root@net:~#vi /etc/cron.daily/logs

0 0 * * * /home/network/log.pl
```

Save and exit.

Execute a job in every 5 seconds

Cron does not provide this feature by default. For this, we need to write up a small bash script to accomplish this task by using the “sleep” command

```
cat seconds.sh
#!/bin/bash
while true
do
    /home/cron/seconds.sh
    sleep 5
done

root@net:~#chmod +x seconds.sh
root@net:~#nohup ./seconds.sh &
```

This command will exit if any error occurred and & signed will put the process in background.

Execute a job in every 4 minutes

If we specify * in the first field, it will run in every minute, it is not the way we want it so we need to add */4 in the along with asterisk. If you wish to run in every 30 min, just add */30

```
root@net:~#vi cron.daily/logs-min
*/4 * * * * /home/network/log-min.pl
```

Save and exit.

Execute a job in every 4 hours

If we specify * in the second field, it will run in every hour; this is not what we want it, so we need to add */4 along with asterisk. If you wish to run in every 15 hours, just add */15

```
root@net:~#vi cron.hourly/logs-hour
* */4 * * * /home/network/log-hourly.pl
```

Save and exit.

Execute a job in every 4th weekdays

The fifth field is DOW (day of the week). If we specify * in the fifth field, it will run in every day. So we need to specify the specific day on which we want to run schedule. In the example, we want to run schedule on every Thursday.

```
root@net:~#vi cron.week/logs-week
* * * * 4 /home/network/log-week.pl

OR

* * * * Thu /home/network/log-week.pl
```

Save and exit.

Execute a job in every 4 months

The third field is DOM (day of the month). If we specify * in the third field, it will run in every day of month. So we need to specify the specific day on which we want to run schedule. The fourth field is for month; If we specify * in the fourth field, it will run in every month. So we need to specify the specific day and month on which we want to run schedule. In the example, we want to run schedule on every first day of oct.

```
root@net:~#vi cron.week/logs-week
* * 1 4 * /home/network/log-month.pl
```

OR

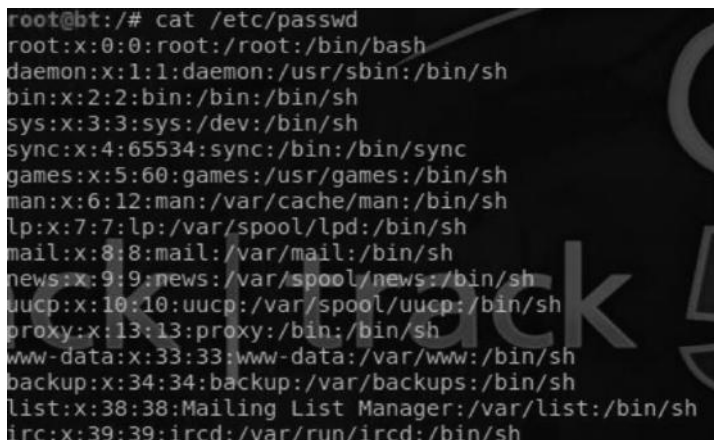
```
* * 1 apr * /home/network/log-month.pl
```

Save and exit.

Note: If you want to assign a range like Jan to Nov then you will need to specify month as 1–11 .

Users inside of Linux

Let's talk about users inside of Linux. The users inside of Linux are stored inside the /etc/passwd file. So here is what the contents of the /etc/passwd file look like:



```
root@bt:/# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
```

So, let's try to understand what the sample entry means. The output for the first line looks like this:

```
root:x:0:0:root:/root:/bin/bash
```

- The “root” is the username.
- The root is followed by x, which means that the password is moved inside the shadow file, which we will discuss next.
- Next is the UID of the user, which is (0) for root, followed by the groupid (0) primary group the user belongs to. In this case, the user belongs to root.
- Next is the space for comments, which an administrator may want to store.
- It is then followed by the absolute path of the home directory, which is also the starting location of the command line.

More about the `/etc/passwd` file:

- In a standard `/etc/passwd` file, most of the users would be default users like `bin/adm` and `mail`.
- All the Unix/Linux users are identified by a user id, which starts at 0 and increments from there with some jumps in between. Any user with uid 0 has root level privileges.
- The nondefault users generally have UIDs starting from 500 or 1000, and increment from there.
- Inside of the `/etc/passwd` file, some users would have `/false` at the end, which means that those users cannot have an interactive login session.

```
snort:x:107:115:Snort IDS:/var/log/snort:/bin/false
statd:x:108:65534::/var/lib/nfs:/bin/false
usbmux:x:109:46::/home/usbmux:/bin/false
pulse:x:110:116::/var/run/pulse:/bin/false
```

Linux Services

The traditional Linux services are inside the `/etc/init.d` directory; this would include scripts to execute a particular service or program that would begin when Linux starts loading.

```
root@bt:/etc/init.d# ls
alsa-mixer-save      hwclock-save        rinetd
apache2              idmapd              rsync
apparmor             irqbalance          rsyslog
appport              killprocs           screen-cleanup
atd                  lm-sensors          sendsigs
avahi-daemon         metasploit-postgres single
binfmt-support       module-init-tools   skeleton
bootlogd             mysql               snort
bridge-network-interface networking           ssh
casper               network-interface   start-yppbind
```

Linux Password Storage

The password for Unix/Linux is stored inside the `/etc/passwd` file or `/etc/shadow` file. Modern Unix-based systems only store passwords in the `/etc/shadow` file and are only readable by root. In older Unix versions, you may find passwords being stored in the `/etc/passwd` file. This is what the `/etc/shadow` file looks like:

```
root@bt:/# cat /etc/shadow
root:$6$BZenJFhs$Qe4sv0CrJHMQ9mmRDuUGjTVllCDQ8qJ/hGwzeaKGTpTx/xU4zp7X8ipcHG6YSAD
HbDuxySnK1PLhK5d1WGpv6/:15920:0:99999:7:::
daemon:x:15907:0:99999:7:::
bin:x:15907:0:99999:7:::
```

The username is followed by a hash. The hashing method would depend upon the version of Linux you are using. MD5 is the most common hashing format for Linux; the password is salted, making it very difficult to crack. You would learn more about cracking password hashes in later parts of this book..

Linux Logging

Now, let's talk briefly about where the log files are stored. The log files are an area of interest for hackers because they want to remove traces of their presence when they have compromised the servers.

Generally the logs are stored inside the `/var/log` and `/var/adm` directory. However, many services such as `httpd` have their own place for storing logs. The Linux saves `.bash_history` inside of the `/home` directory. The `.bash_history` file contains list of commands that were used from `bash`.

Common Applications of Linux

Here are some of the common applications that you would most probably encounter with any Linux flavor you use:

- *Apache*—This is an open source web server. Most of the web runs on the Apache web server.
- *MySQL*—This is the most popular database used in Unix-based systems.
- *Sendmail*—This is a free Linux-based mail server. It is available inside both open source and commercial versions.
- *Postfix*—This can be used as a send-mail alternative.
- *PureFTP*—This is the default ftp server used for almost all Unix-based systems.
- *Samba*—This provides file and printer sharing services. The best part is that it can easily integrate with Windows-based systems.

What Is BackTrack?

So now that you are familiar with Linux, let me introduce you to BackTrack. BackTrack is a Linux penetration testing distro developed by Offensive Security especially for ethical hackers and penetration testers. It contains all the popular tools and software used for pen testing a variety of services, networks, and devices.

BackTrack 5 is the latest version of the Linux penetration testing distro at the time of writing this chapter. It comes in two flavors: Gnome and KDE. Gnome is an Ubuntu-based Linux operating system that has officially been introduced only in the latest version of BackTrack. Here is a screenshot of BackTrack 5.



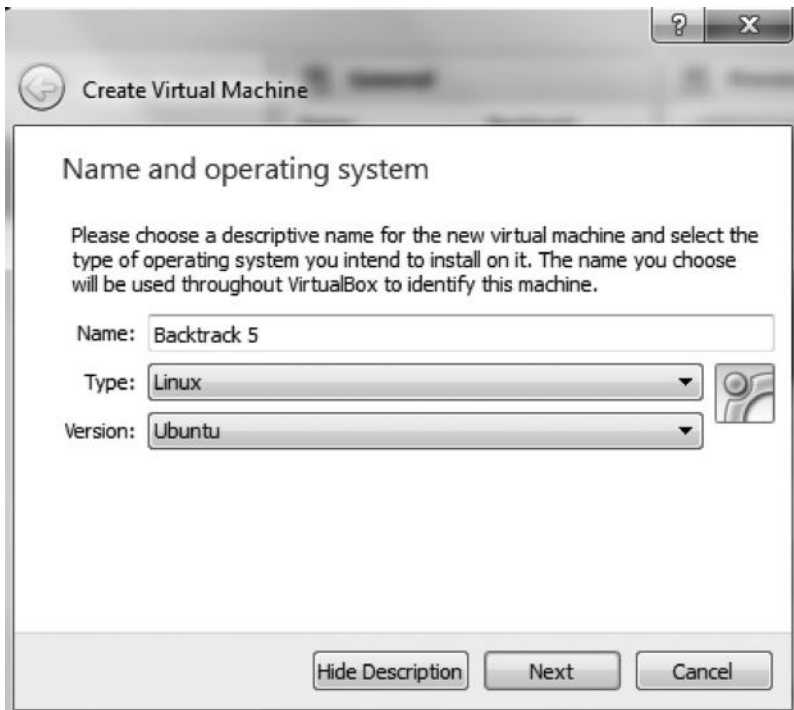
How to Get BackTrack 5 Running

Now that you have a basic idea of what BackTrack is and why it is used, it's time to install BackTrack on our box and get things going. There are many ways you can get BackTrack up and running. I install BackTrack on a virtualization software such as VMware or virtual box. Personally, I am a fan of virtual box, since it does not take much of my computer's memory. Therefore, what we will learn next is how to install BackTrack on virtual box.

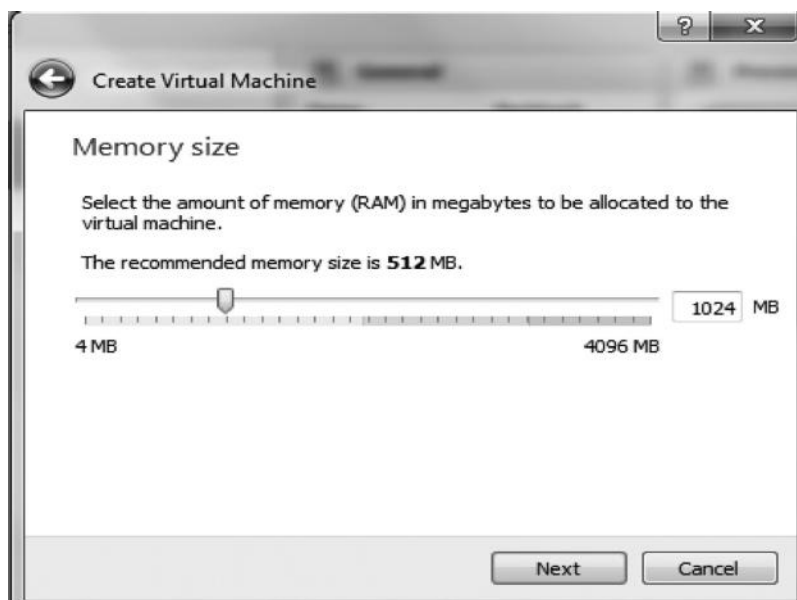
Installing BackTrack on Virtual Box

There are times when we need to switch between operating systems rapidly and we need our BackTrack running alongside another OS like Windows or Red Hat Linux. One advantage of doing this is it gives us more accessibility. For doing this you need to download VM Virtual Box, which is a freely available tool.

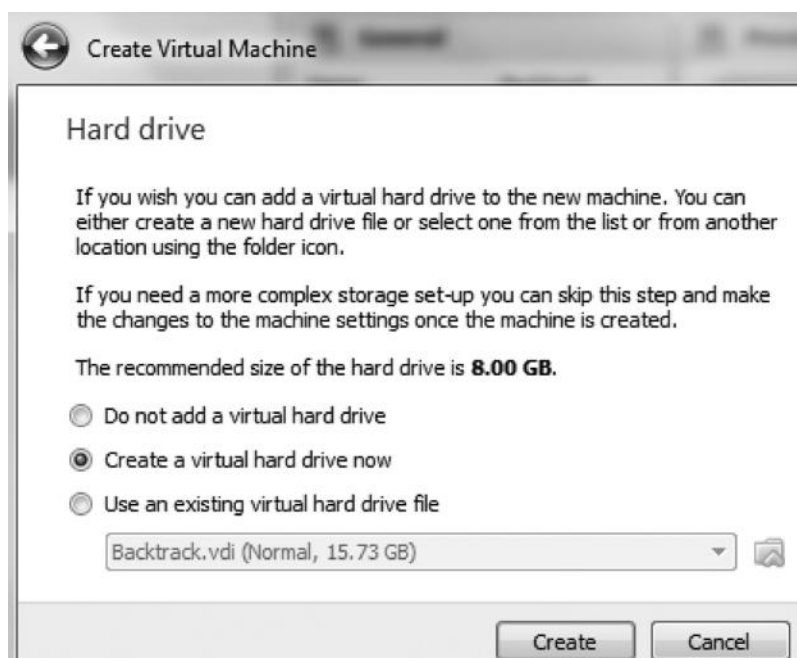
Step 1—After downloading and installing virtual box on to your PC, click on the “New” button. A dialogue box will appear where you would need to type the name of the “OS,” the “Version,” and the operating system type. In my case the name would be “BackTrack,” the OS “Linux,” and the version “Ubuntu.”



Step 2—The next step would be to allocate the RAM; it is recommended that you allocate at least 1024 MB (1 GB) for BackTrack to run perfectly.



Step 3—Next, choose to create a virtual drive and then in the next window select the hard drive type as VDI (Virtual Disk Image).



Hard drive file type

Please choose the type of file that you would like to use for the new virtual hard drive. If you do not need to use it with other virtualization software you can leave this setting unchanged.

- ☒ VDI (VirtualBox Disk Image)
- ☐ VMDK (Virtual Machine Disk)
- ☐ VHD (Virtual Hard Disk)
- ☐ HDD (Parallels Hard Disk)
- ☐ QED (QEMU enhanced disk)
- ☐ QCOW (QEMU Copy-On-Write)

Step 4—In the next step, you have to choose if you want the hard disk to be dynamically allocated or have a fixed size. If you have enough space on your hard disk, you might want to choose the first option. Nevertheless, it's up to you.



Create Virtual Hard Drive

Storage on physical hard drive

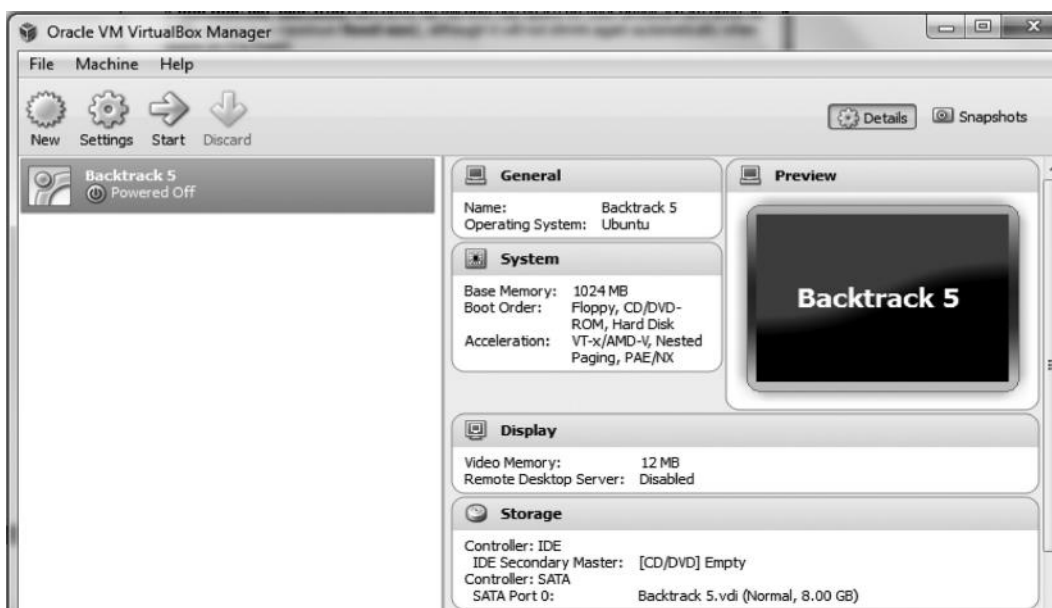
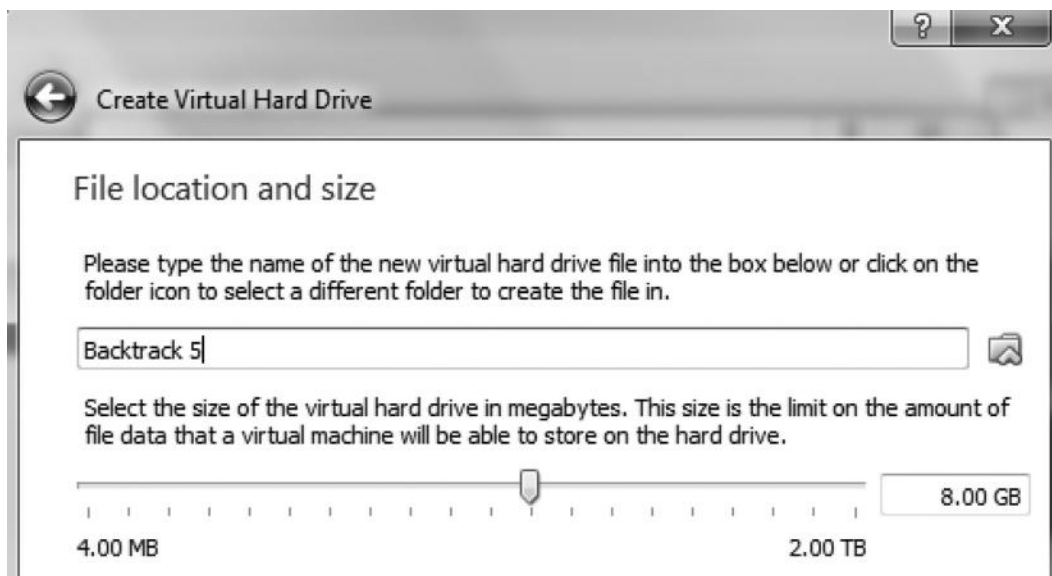
Please choose whether the new virtual hard drive file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard drive file will only use space on your physical hard drive as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

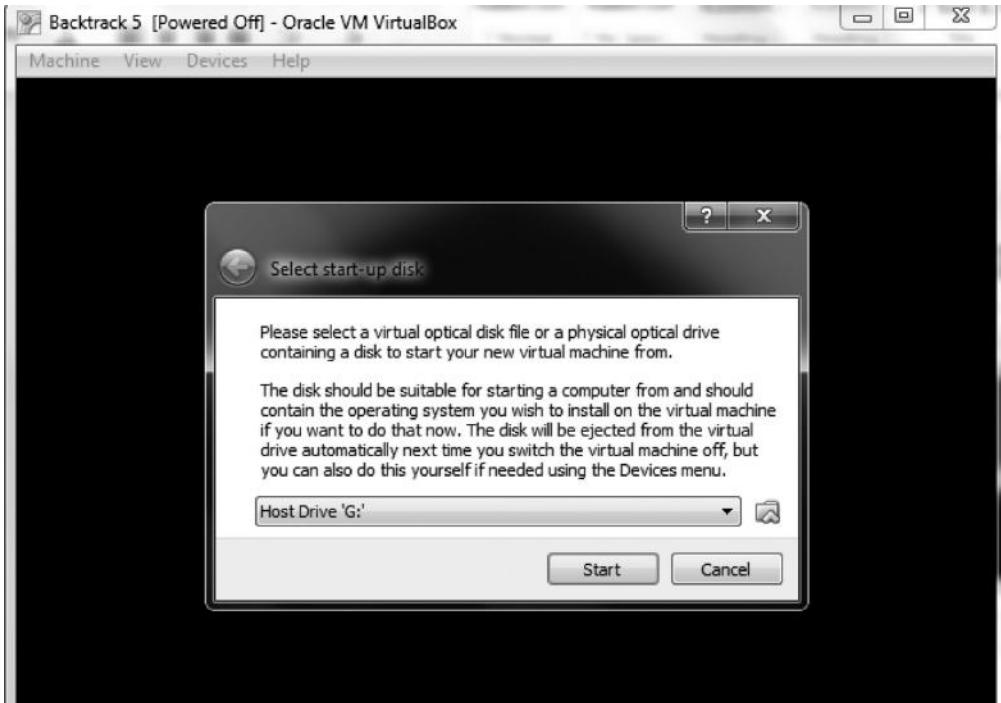
A **fixed size** hard drive file may take longer to create on some systems but is often faster to use.

- ☒ Dynamically allocated
- ☐ Fixed size

Step 5—Next, choose the name of your virtual hard drive and allocate the size of the hard disk.



Step 6—So, now when the virtual hard disk has been created and other settings are selected, load the BackTrack that was downloaded onto the virtual box and click “Start”.



That's all we need to do. We now have BackTrack installed on our virtual box.

Installing BackTrack on a Portable USB

BackTrack can also be made portable by installing it on to a USB flash drive. This way you can carry BackTrack Live anywhere. This practice is useful for outsource penetration tests and, moreover, it is very easy to make BackTrack USB.

For this you need the following:

- USB flash drive (minimum 8 GB)
- A disk burning software

For this purpose, we are going to use PowerISO, which is freely available online at <http://www.poweriso.com>

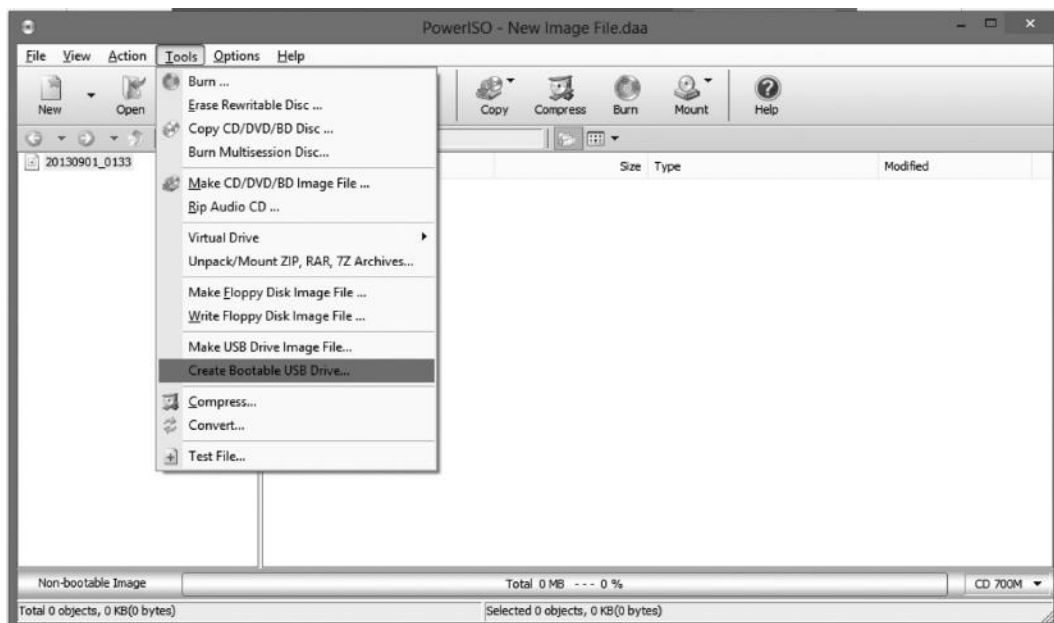
Step 1—Format your flash drive and *ensure* that it has at least 7 GB of free space.



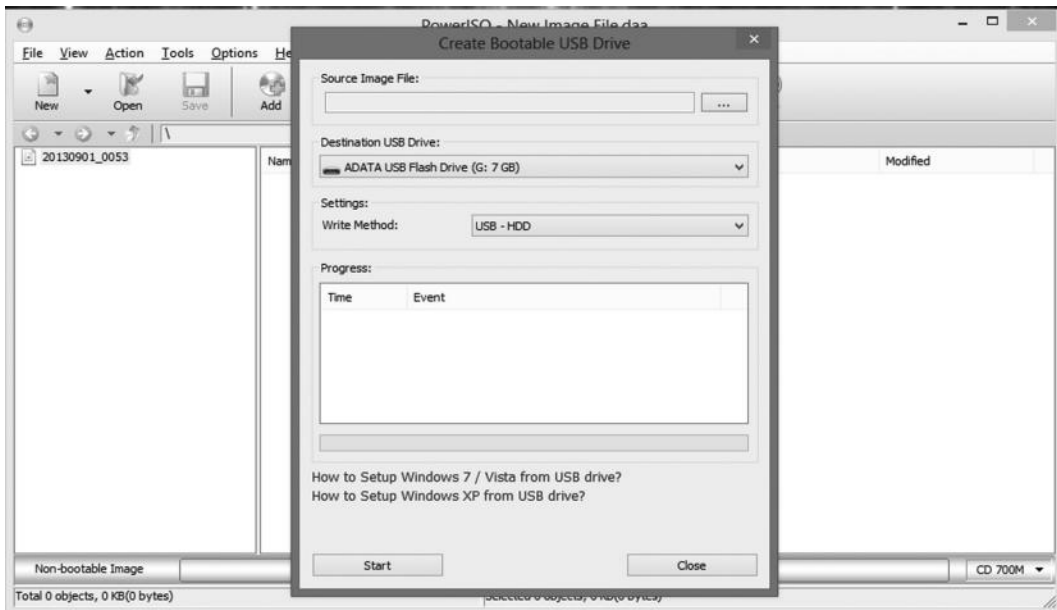
Step 2—Open PowerISO from the “Start” menu.



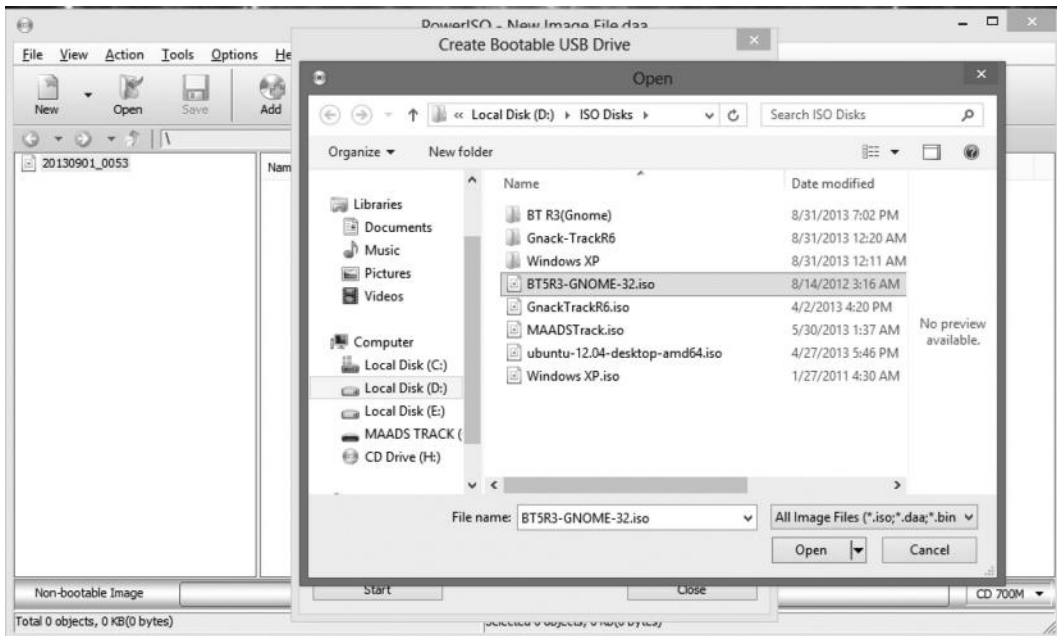
Step 3—Click on “Tools” and from *the* dropdown list select “Make a bootable USB.”



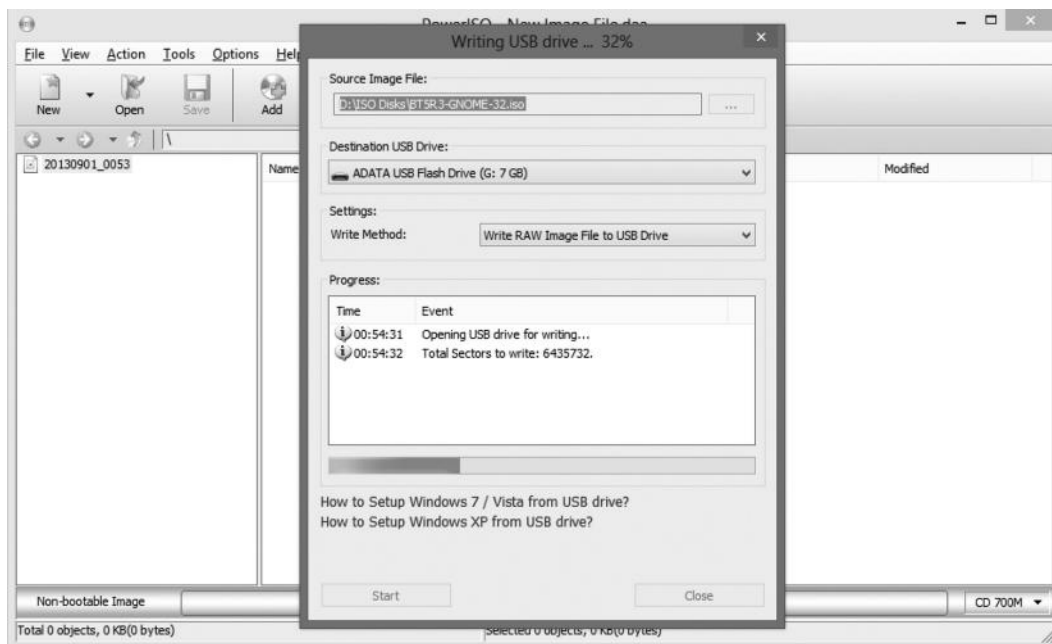
Step 4—The following dialogue box will appear.



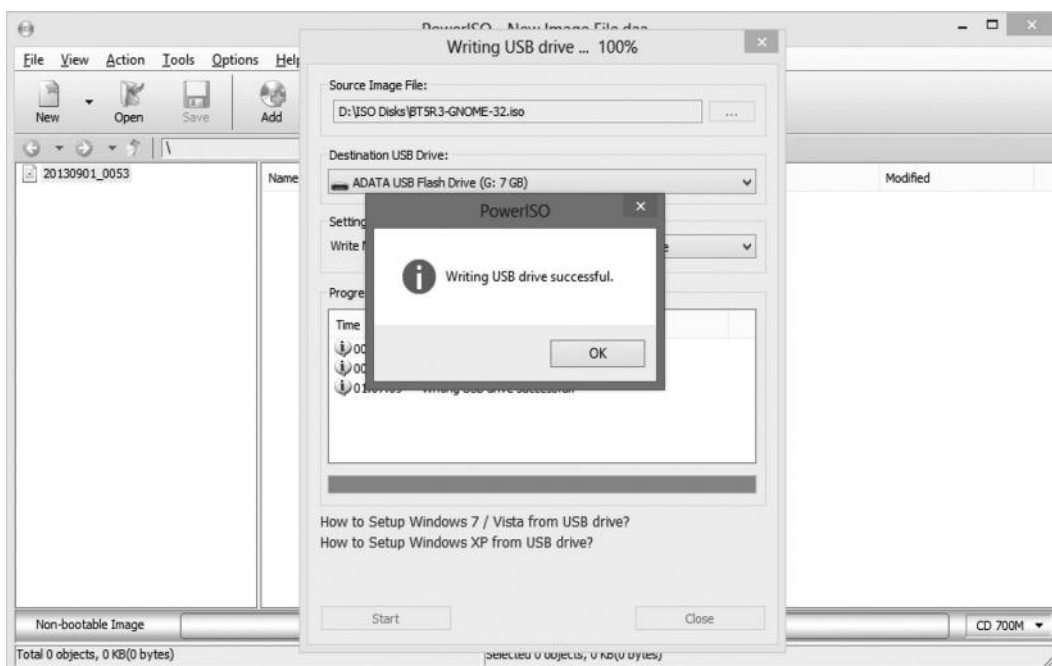
Step 5—Locate your BackTrack ISO disk image.



Step 6—Now it will start burning the image on to your USB drive.



Step 7—When the process is complete, the following message appears.



Installing BackTrack on Your Hard Drive

If you run BackTrack from VMware or virtual box, any changes you made would be removed after rebooting; to solve this issue, we need to install BackTrack on the hard drive.

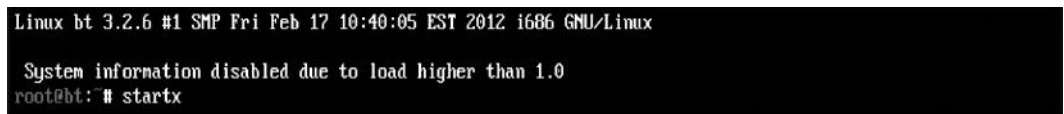
For this, we need two things:

1. BackTrack Live CD or BackTrack installed on VMware or virtual box.
2. A hard drive with minimum 20 GB free space.

Step 1—Insert the disk into the drive and boot from it. This is what you will see in the beginning:



Step 2—Then you will see the screen `root@bt:`, where you will have to type the command “`startx`”.



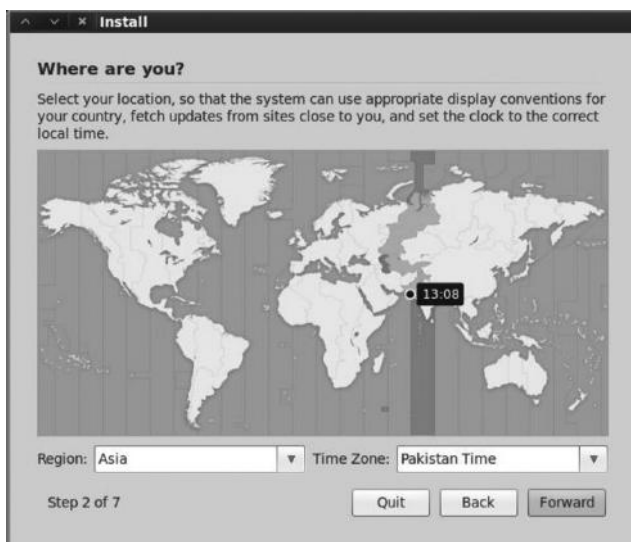
Step 3—Now that we have booted into BackTrack, we will install it on our hard drive. Click on the icon “Install BackTrack” and your installation should start.



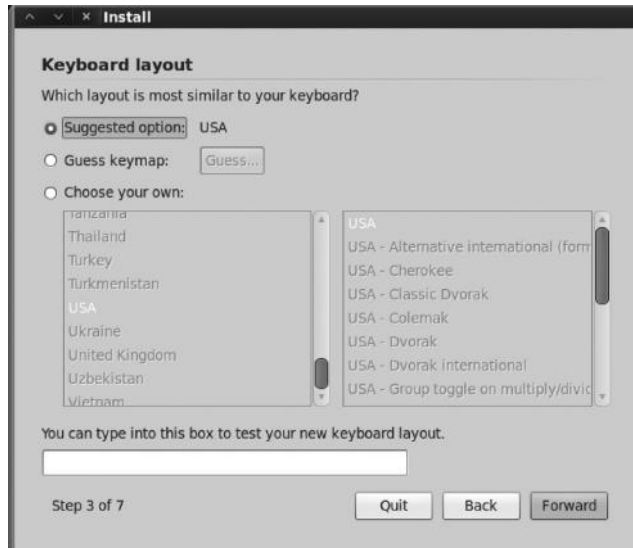
Step 4—On the Welcome screen, you will have to select the appropriate language and click “Forward”.



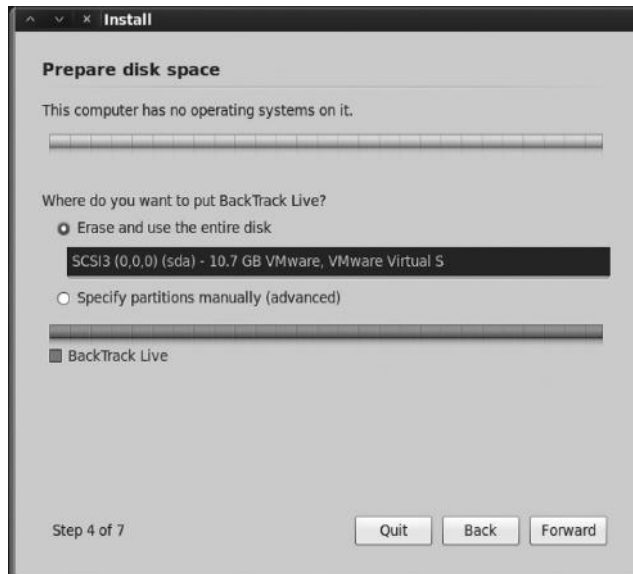
Step 5—Now select your time zone. Or, if you are already connected to the network, your time zone will automatically be detected.



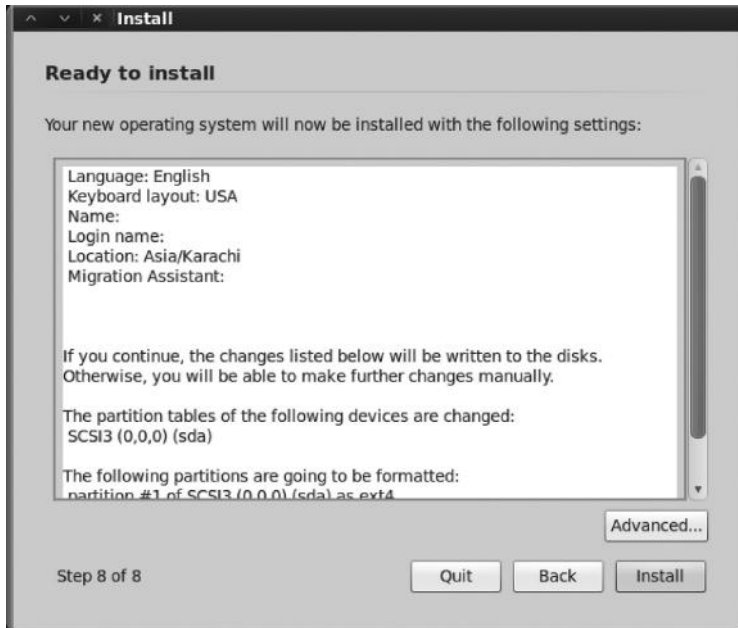
Step 6—Now a window to select the desired keyboard layout appears.



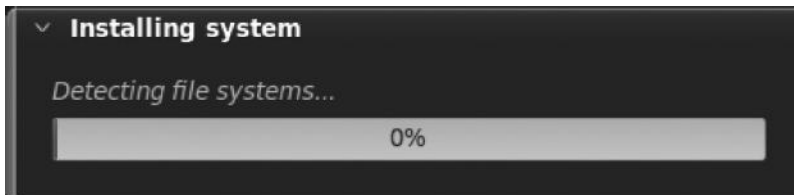
Step 7—Next we will have to set the partition size. In most cases we leave it to default and the entire partition is erased.



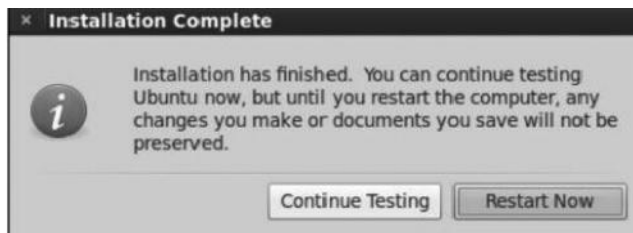
Step 8—Now the install summary appears and you just have to click on “Install” and your work is done.



The installer will take some time to complete, which may be several minutes.



After the installation is complete, you will be prompted to restart your PC and as you reset your BackTrack, it will be installed to your hard drive.



BackTrack Basics

Once you have BackTrack up and running, it's time to learn about BackTrack basics. By the time you are reading this book, BackTrack would have been upgraded to version 6 or 7, and you might be wondering if the techniques discussed work only for BackTrack 5. If so, then you are wrong.

Starting from BackTrack 1 all the way to BackTrack 5, the only thing that changed were the tools. Outdated tools are removed and new tools are added, but the structure and fundamentals stay the same.

One of the common problems I see with beginners is that they tend to use the KDE menu a lot. I suggest you stay away from the KDE menu and try to use the command line before jumping to the KDE menu. I want you to familiarize yourself with BackTrack's environment as it will be discussed in many of the upcoming chapters, especially in the later chapters of this book.

Taking you back to BackTrack, the `/pentest` directory is by far the most important directory present in BackTrack as it has all the penetration testing tools. To access the pentest directory of BackTrack, open up your shell and type "`cd/pentest`" and then type "`ls`". "`ls`" will get you into all the subdirectories present in the pentest directory.



```

pentest: bash
File Edit View Bookmarks Settings Help
root@bt: # cd /pentest
root@bt:/pentest# ls
backdoors database forensics misc re rfid stressing voip wireless
bluetooth enumeration fuzzers passwords reporting scanners telephony web
cisco exploits libs python reverse-engineering sniffers tunneling windows-binaries
root@bt:/pentest#

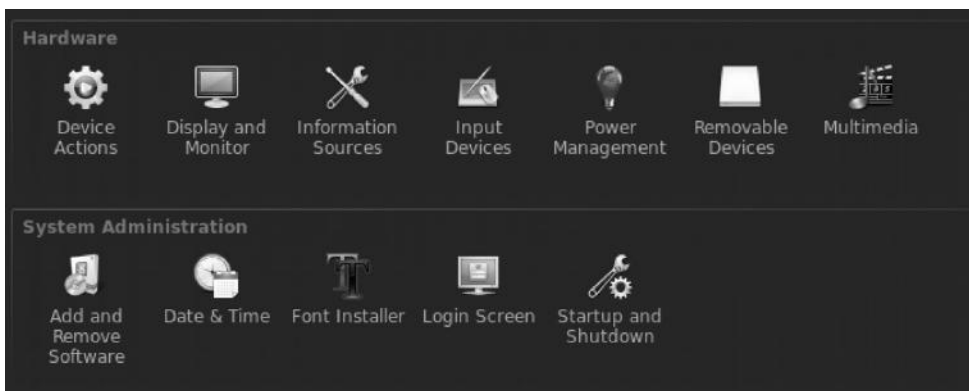
```

Changing the Default Screen Resolution

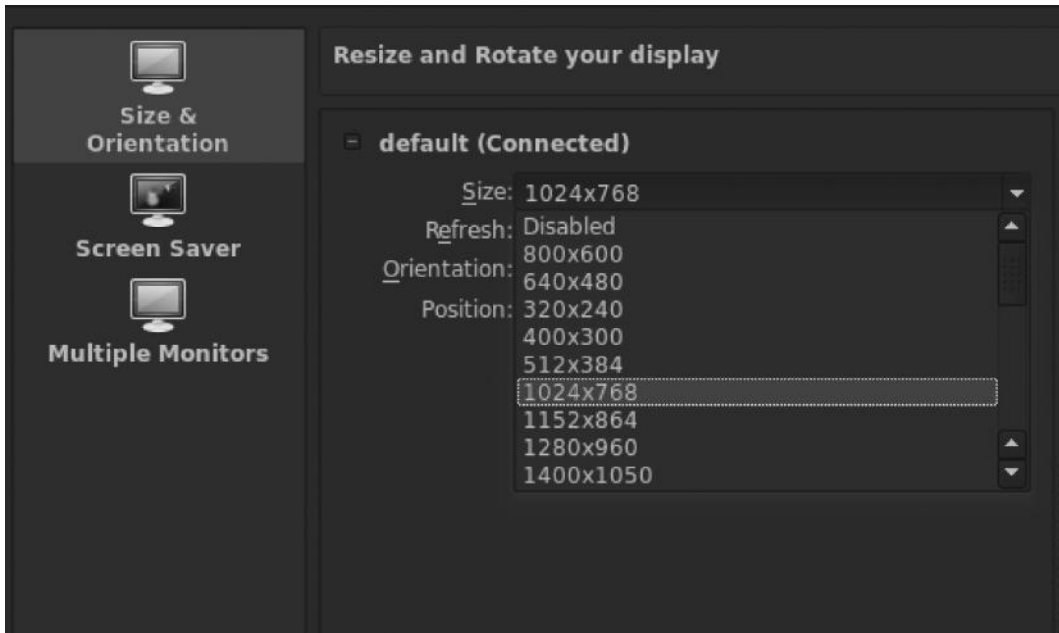
The default size of the BackTrack 5 screen is 800 by 600, which is very small and is not recommended. If you want to change your BackTrack 5 (KDE) default screen size, then just follow these steps:

Step 1—Go to Start → Settings → System Settings

Step 2—Then from the hardware section click on “Display and Monitor”



Step 3—Next choose your preferred size and click “Ok”. A dialog box will now appear asking you to confirm the changes. Just click “Accept Configuration” and you are done.



Some Unforgettable Basics

Changing the Password

We would need to issue the following command in order to change the password of our Linux box. Generally, it's a good practice to change the default password to prevent unscrupulous people from getting into the network. This is the reason I have kept this command at the top of the basics list.

```
passwd
```

Clearing the Screen

In Windows command prompt we use “cls”; inside Linux BackTrack we use the `clear` command.

Listing the Contents of a Directory

```
ls
```

`ls` is used for listing the contents in a directory, the `-l` parameter can also be used for listing the permissions of the current directory.

Displaying Contents of a Specific Directory

```
ls/pentest/enumeration
```

It is used to list the contents of a specific directory. Issuing this command generates a list of the contents of the `/pentest/enumeration` directory.

Displaying the Contents of a File

```
cat password.txt
```

This command lists the contents of the passwords file.

Creating a Directory

```
mkdir directoryname
```

The process is the same as in Windows.

Changing the Directories

```
cd/pentest/enumeration
```

Changing the directories is very simple. It works as in Windows. However, we use / in Linux instead of \ for changing the directories.

Windows

```
C:/windows/settings
```

Linux

```
/pentest/web/scanners
```

Creating a Text File

```
touch hack.txt
```

This command creates a text file with the name hack.txt.

Copying a File

```
Cp source target
```

```
cp /var/www/filename /pentest/web/filename
```

This command will copy the file from the /var/www directory to the /pentest/web/ directory.

Current Working Directory

```
pwd
```

This will return the current working directory.

Renaming a File

```
mv oldfile.txt newfile.txt
```

There is no command specifically for renaming files inside Linux; however, you just need to issue the mv command to rename the file.

Moving a File

```
mv hack.txt/pentest/enumeration/
```

This command will move the file hack.txt to the /pentest/enumeration directory.

Removing a File

```
rm file name
```

This is very simple, and it works for directories in the same way.

Locating Certain Files inside BackTrack

Let's say we are searching for "TheHarvester" tool and we don't know in which directory it exists. We can use the locate command to find it.

Example

```
locate harvester
```



```
/pentest/enumeration/theharvester/hostchecker.pyc
/pentest/enumeration/theharvester/parser.py
/pentest/enumeration/theharvester/parser.pyc
/pentest/enumeration/theharvester/theHarvester.py
/pentest/enumeration/theharvester/version.txt
/pentest/enumeration/theharvester/discovery/__init__.py
/pentest/enumeration/theharvester/discovery/__init__.pyc
/pentest/enumeration/theharvester/discovery/bingsearch.py
/pentest/enumeration/theharvester/discovery/bingsearch.pyc
/pentest/enumeration/theharvester/discovery/exaleadsearch.py
/pentest/enumeration/theharvester/discovery/exaleadsearch.pyc
/pentest/enumeration/theharvester/discovery/googlesearch.py
/pentest/enumeration/theharvester/discovery/googlesearch.pyc
/pentest/enumeration/theharvester/discovery/linkedinsearch.py
/pentest/enumeration/theharvester/discovery/linkedinsearch.pyc
/pentest/enumeration/theharvester/discovery/pgpsearch.py
/pentest/enumeration/theharvester/discovery/pgpsearch.pyc
```

Text Editors inside BackTrack

BackTrack by default does not have any fancy text editors like Notepad in Windows. It has some text editors that we can use within the command line such as nano, pico, and vim.

However, if you want to use a text editor that is equivalent to Notepad in Windows, I would recommend you use kate or gedit.

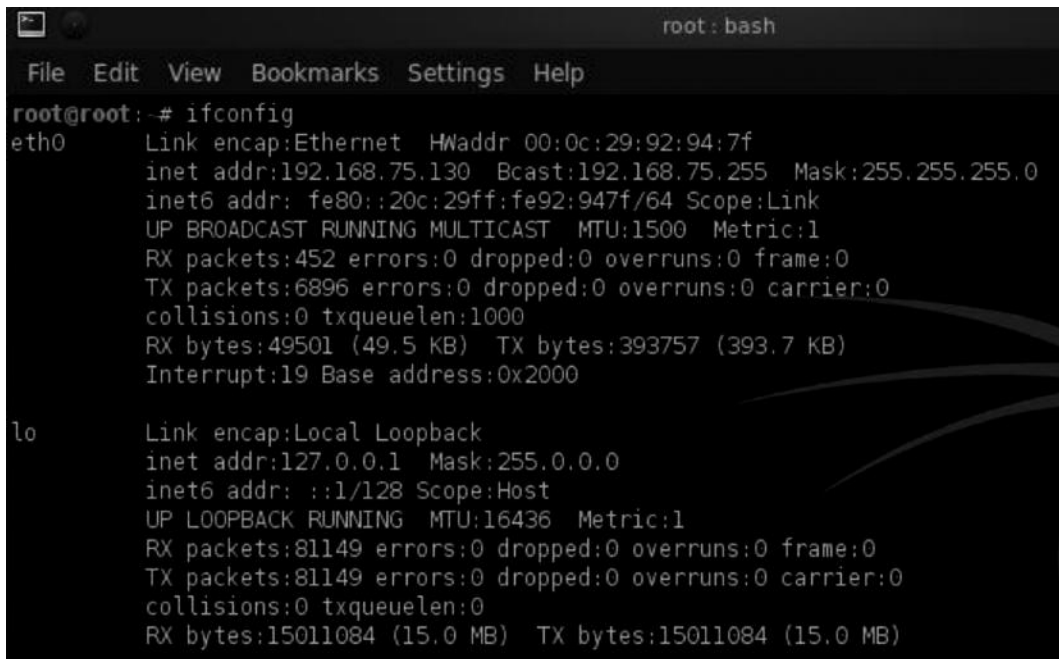
In order to install them, you would need to issue the following commands from the command line:

```
apt-get install gedit
apt-get install kate
```

These commands will automatically search the Internet and download the packages and dependencies.

Getting to Know Your Network

The first thing that we need to check when we are on BackTrack is that if we have a valid IP address. If you type the command “ifconfig” in your command line, it will list all of your current configurations.



```
root@root:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:92:94:7f
          inet addr:192.168.75.130  Bcast:192.168.75.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe92:947f/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:452 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6896 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:49501 (49.5 KB)  TX bytes:393757 (393.7 KB)
          Interrupt:19 Base address:0x2000

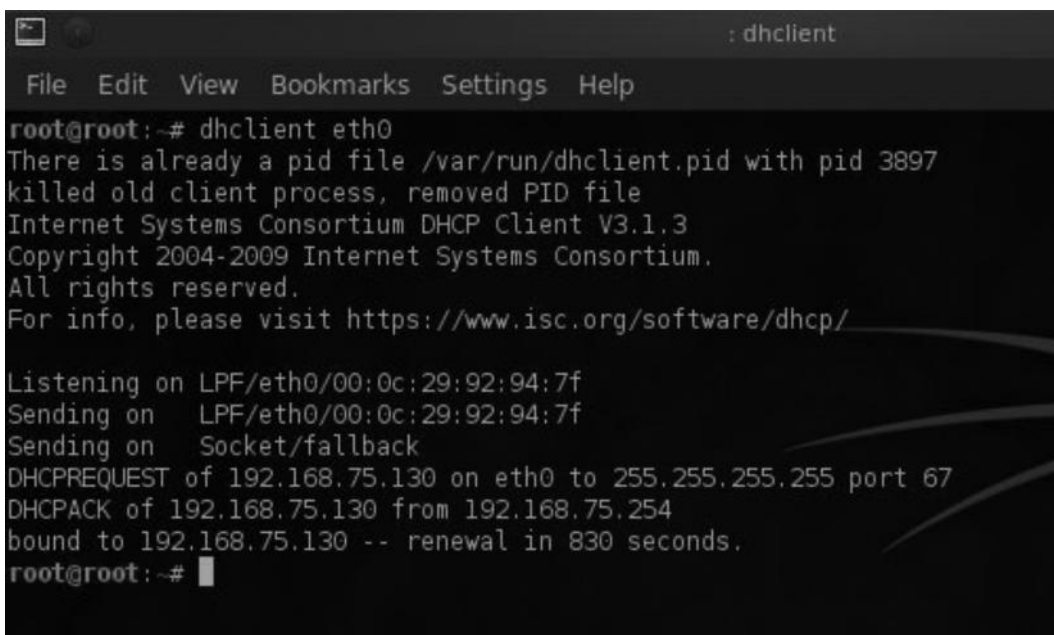
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:81149 errors:0 dropped:0 overruns:0 frame:0
          TX packets:81149 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:15011084 (15.0 MB)  TX bytes:15011084 (15.0 MB)
```

As you can see from the screenshot, the local IP is 192.168.75.130 and the subnet mask is 255.255.255.0; you can also see other configurations including network interfaces.

Dhclient

By running the command Dhclient followed by the interface on the terminal, a new static IP address will automatically be assigned by DHCP. However, if for any reason this method does not work for you, you can start networking by issuing the following command:

```
root@bt:~# /etc/init.d/networking start
```



```
File Edit View Bookmarks Settings Help
root@root:~# dhclient eth0
There is already a pid file /var/run/dhclient.pid with pid 3897
killed old client process, removed PID file
Internet Systems Consortium DHCP Client V3.1.3
Copyright 2004-2009 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:0c:29:92:94:7f
Sending on   LPF/eth0/00:0c:29:92:94:7f
Sending on   Socket/fallback
DHCPREQUEST of 192.168.75.130 on eth0 to 255.255.255.255 port 67
DHCPACK of 192.168.75.130 from 192.168.75.254
bound to 192.168.75.130 -- renewal in 830 seconds.
root@root:~#
```

Services

BackTrack has a variety of useful services such as Apache and MySQL that are disabled by default. You can enable these services by issuing various commands on your console.

Note: Before starting any services such as SSH, you should consider changing your root password, which is “toor” by default to prevent hackers and other unscrupulous people to get into your network.

MySQL

By default the MySQL service runs in your BackTrack 5 OS. You can easily start or stop the service by issuing the following init.d script:

Start—`/etc/init.d/mysql start`

Stop—`/etc/init.d/mysql stop`

SSHD

SSH functions the same way as the FTP protocol. However, it is used for secure file sharing as the data being sent and received is encrypted. So it's considered more secure than ftp. However, weaknesses have also been identified in SSHD clients though it's relatively more secure than FTP.

In order to start an SSH server, first you need to generate SSH keys. You can generate SSH keys by simply issuing the following command in your console.

```

root@root:~# sshd-generate
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
e8:b9:c7:e7:7d:3c:97:39:6f:a3:a1:ab:90:be:de:d1 root@root
The key's randomart image is:
+--[RSA1 2048]-----+
|
|      .
|     . S
|    . . .
|   o+ . E o o|
|  ..= o.. ==o|
|   o=..0000.+|=
+-----+
Generating public/private rsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_rsa_key.
Your public key has been saved in /etc/ssh/ssh_host_rsa_key.pub.
The key fingerprint is:
14:45:7e:81:fa:45:3a:86:62:1c:b7:13:c4:94:07:55 root@root
The key's randomart image is:
+--[RSA 2048]-----+

```

Let's now connect to your SSH server from your Windows operating system. In order to do that you would need an SSH client such as putty.

Step 1—Run the following command in order to start the SSH server on your BackTrack.

```
/etc/init.d/ssh start
```

You can verify if SSH is running by typing the following command:

```
netstat -ano | grep 22
```

```

root@root:~# netstat -ano | grep 22
tcp        0      0 0.0.0.0:22 0.0.0.0:*        LISTEN      off (0.00/0.0)
tcp6       0      0 :::22     :::*              LISTEN      off (0.00/0.0)
unix  2      [ ]          DGRAM                    5225
root@root:~# _

```

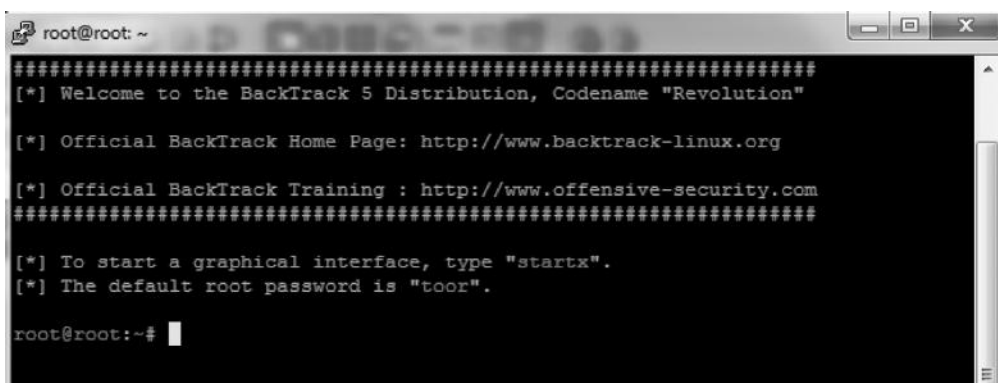
Next, type “ifconfig” from your terminal to obtain your IP address.

Step 2—Open up putty on your Windows operating system. Type your BackTrack IP address and connect to port 22.



Step 3—Now it will ask you for your credentials. Enter “root” as username and “toor” as password in case you haven’t changed the default credentials.

Step 4—Once you have entered the credentials, you will be inside the BackTrack console; now you can run BackTrack from your Windows.



Postgresql

By default, BackTrack 5 box does not come with postgresql. However, Metasploit does support postgresql databases. In order to install postgresql, we need to issue the following command in the console.

```
apt-get install postgresql
```


Once postgresql is successfully installed on your BackTrack 5 box, all you need to do is issue the following service init script in order to start the postgresql service.

```
/etc/init.d/postgresql start
```

However, if you are still facing problems in getting postgresql up and running, don't worry. We shall get to it once we reach the "Remote exploitation" chapter of this book.

BackTrack 5 also offers a wide variety of other services, such as tftpd and apache, which you can also run from the command line and which are also present in the KDE menu. The services are present in the BackTrack → Services tab in the main menu.



Other Online Resources

- <http://Linux.org>
- <http://beginLinux.org>
- <http://Linux-tutorial.info>
- BackTrack-Linux.org

Requiring no prior hacking experience, *Ethical Hacking and Penetration Testing Guide* supplies a complete introduction to the steps required to complete a penetration test, or ethical hack, from beginning to end. You will learn how to properly utilize and interpret the results of modern-day hacking tools that are required to complete a penetration test.

The book covers a wide range of tools, including Backtrack Linux, Google Reconnaissance, MetaGooFil, dig, Nmap, Nessus, Metasploit, Fast Track Autopwn, Netcat, and Hacker Defender rootkit. Supplying a simple and clean explanation of how to effectively utilize these tools, it details a four-step methodology for conducting an effective penetration test or hack.

Providing an accessible introduction to penetration testing and hacking, the book supplies you with a fundamental understanding of offensive security. After completing the book you will be prepared to take on in-depth and advanced topics in hacking and penetration testing.

The book walks you through each of the steps and tools in a structured, orderly manner allowing you to understand how the output from each tool can be fully utilized in the subsequent phases of the penetration test. This process will allow you to clearly see how the various tools and phases relate to each other.

An ideal resource for those who want to learn about ethical hacking but don't know where to start, this book will help take your hacking skills to the next level. The topics described in this book comply with international standards and with what is being taught in international certifications.



CRC Press
Taylor & Francis Group
an **informa** business
www.crcpress.com

6000 Broken Sound Parkway, NW
Suite 300, Boca Raton, FL 33487
711 Third Avenue
New York, NY 10017
2 Park Square, Milton Park
Abingdon, Oxon OX14 4RN, UK

K22730

ISBN: 978-1-4822-3161-8



www.auerbach-publications.com