Kali Linux - An Ethical Hacker's Cookbook

Second Edition

Practical recipes that combine strategies, attacks, and tools for advanced penetration testing

Himanshu Sharma

Packt
www.packt.com

Kali Linux - An Ethical Hacker's Cookbook
Second Edition

Practical recipes that combine strategies, attacks, and tools for advanced penetration testing

Himanshu Sharma

Packt

BIRMINGHAM - MUMBAI

Kali Linux - An Ethical Hacker's Cookbook
Second Edition

Commissioning Editor: Vijin Boricha
Acquisition Editor: Rohit Rajkumar
Content Development Editor: Ronn Kurien
Technical Editor: Prachi Sawant
Copy Editor: Safis Editing
Project Coordinator: Jagdish Prabhu
Proofreader: Safis Editing
Indexer: Manju Arasan
Graphics: Tom Scaria
Production Coordinator: Jayalaxmi Raja

First published: October 2017
Second edition: March 2019

Production reference: 1290319

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham
B3 2PB, UK.

ISBN 978-1-78995-230-8

www.packtpub.com

Contributors

About the author

Himanshu Sharma has been active in the field of bug bounty since 2009, and has been listed in Apple, Google, Microsoft, Facebook, Adobe, Uber, AT&T, Avira, and many more with hall of fame listings as proof.

He has been a speaker at multiple international conferences, including Botconf '13, Confidence 2018, RSA Asia Pacific and Japan '18, and Hack In The Box 2019. He also spoke at the IEEE conference in California and Malaysia, as well as for TedX.

Currently, he is the cofounder of BugsBounty, a crowd-sourced security platform for ethical hackers and companies interested in cyber services. He has also authored the following books: *Kali Linux – An Ethical Hacker's Cookbook*, and *Hands-On Red Team Tactics*.

About the reviewers

Bhargav Tandel has over 7 years' experience in information security with companies including Reliance jio, Vodafone, and Wipro. His core expertise and passions are vulnerability assessment, penetration testing, Red Team, ethical hacking, and information security. He is currently pursuing the OSCP certification. He has the ability to solve complex problems involving a wide variety of information systems, work independently on large-scale projects, and thrive under pressure in fast-paced environments, all while directing multiple projects from concept to implementation.

I would like to thank my family and friends, who have always stood by me. My friends, Jigar Tank and Utkarsh Bhatt, have always been there for me. I would also like to thank Rakesh Dwivedi for giving me a reason to continue learning and growing.

Kunal Sehgal has been heading critical cybersecurity roles for financial organizations, for over 15 years now. He is an avid blogger and a regular speaker on cyber-related topics across Asia.

He also holds a bachelor's degree in computer applications from Panjab University, and a postgraduate diploma from Georgian College in cyberspace security. He holds numerous cyber certifications, including Certified Information Systems Auditor (CISA), Certified Information Systems Security Professional (CISSP), Certified Information Security Manager (CISM), Tenable Certified Nessus Auditor (TCNA), Certificate of Cloud Security Knowledge (CCSK), ISO 27001 Lead Auditor, Offensive Security Certified Professional (OSCP), and CompTIA Security+.

Dedicated to my darling daughter.

Shivanand Persad has a master's in business administration from the Australian Institute of Business, and a bachelor of science in electrical and computer engineering from the University of the West Indies. He possesses a wide variety of specializations, including controls and instrumentation systems, wireless and wired communication systems, strategic management, and business process re-engineering. With over a decade of experience across multiple engineering disciplines, and a lengthy tenure with one of the largest ISPs in the Caribbean, he continues to be passionate about technology and its continuous development. When he's not reading everything in sight, he enjoys archery, martial arts, biking, and tinkering.

Packt is searching for authors like you

If you're interested in becoming an author for Packt, please visit `authors.packtpub.com` and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Table of Contents

Preface 1

Chapter 1: Kali - An Introduction 6
- **Configuring Kali Linux** 6
  - Getting ready 7
  - How to do it... 8
  - How it works... 8
- **Configuring the Xfce environment** 8
  - How to do it... 8
- **Configuring the MATE environment** 11
  - How to do it... 11
- **Configuring the LXDE environment** 12
  - How to do it... 12
- **Configuring the E17 environment** 14
  - How to do it... 14
- **Configuring the KDE environment** 14
  - How to do it... 14
- **Prepping with custom tools** 16
  - Getting ready 16
  - How to do it... 16
    - Aquatone 16
    - Subfinder 19
  - There's more... 22
- **Zone Walking using DNSRecon** 23
  - Getting ready 23
  - How to do it... 23
  - There's more... 25
- **Setting up I2P for anonymity** 26
  - How to do it... 26
  - There's more... 29
- **Pentesting VPN's ike-scan** 29
  - Getting ready 30
  - How to do it... 30
    - Cracking the PSK 32
  - There's more... 32
- **Setting up proxychains** 33
  - How to do it... 33
    - Using proxychains with Tor 34
- **Going on a hunt with Routerhunter** 35
  - Getting ready 35

How to do it... 36
Chapter 2: Gathering Intel and Planning Attack Strategies 37
Getting a list of subdomains 38
How to do it... 38
Using Shodan for fun and profit 39
Getting ready 40
How to do it... 40
Shodan Honeyscore 43
How to do it... 44
Shodan plugins 45
How to do it... 45
Censys 46
How to do it... 46
See also 51
Using Nmap to find open ports 52
How to do it... 52
Using scripts 53
See also 54
Bypassing firewalls with Nmap 55
How to do it... 55
TCP ACK scan (-sA) 55
TCP Window scan (-sW) 56
Idle scan 56
How it works... 57
Searching for open directories using GoBuster 57
How to do it... 58
Hunting for SSL flaws 60
How to do it... 60
See also 62
Automating brute force with BruteSpray 62
How to do it... 62
Digging deep with TheHarvester 64
How to do it... 64
How it works... 65
Finding technology behind webapps using WhatWeb 66
How to do it... 66
Scanning IPs with masscan 67
How to do it... 67
Finding origin servers with CloudBunny 68
How to do it... 68
Sniffing around with Kismet 70
How to do it... 71
See also 75
Testing routers with Firewalk 75

How to do it... 76
How it works... 77

Chapter 3: Vulnerability Assessment - Poking for Holes 78
Using the infamous Burp 79
How to do it... 79
Exploiting WSDLs with Wsdler 87
How to do it... 88
Using Intruder 91
How to do it... 91
Using golismero 95
How to do it... 96
See also 98
Exploring Searchsploit 98
How to do it... 99
Exploiting routers with routersploit 100
Getting ready 100
How to do it... 100
Using Metasploit 103
How to do it... 104
Automating Metasploit 106
How to do it... 107
Writing a custom resource script 108
How to do it... 109
See also 109
Setting up a database in Metasploit 110
How to do it... 110
Generating payloads with MSFPC 113
How to do it... 113
Emulating threats with Cobalt Strike 117
Getting ready 117
How to do it... 119
There's more... 130

Chapter 4: Web App Exploitation - Beyond OWASP Top 10 133
Exploiting XSS with XSS Validator 134
Getting ready 134
How to do it... 135
Injection attacks with sqlmap 140
How to do it... 141
See also 143
Owning all .svn and .git repositories 143
How to do it... 143
Winning race conditions 144
How to do it... 145

See also 146
Exploiting XXEs 147
How to do it... 147
See also 150
Exploiting Jboss with JexBoss 151
How to do it... 151
Exploiting PHP Object Injection 153
How to do it... 153
See also 156
Automating vulnerability detection using RapidScan 157
Getting ready 157
How to do it... 157
Backdoors using meterpreter 160
How to do it... 160
See also 163
Backdoors using webshells 163
How to do it... 163
Chapter 5: Network Exploitation 170
Introduction 170
MITM with hamster and ferret 171
Getting ready 171
How to do it... 171
Exploring the msfconsole 173
How to do it... 174
Railgun in Metasploit 177
How to do it... 177
There's more... 180
See also 181
Using the paranoid meterpreter 181
How to do it... 181
There's more... 183
The tale of a bleeding heart 183
How to do it... 183
Exploiting Redis 186
How to do it... 186
Saying no to SQL – owning MongoDBs 189
Getting ready 189
How to do it... 190
Hacking embedded devices 191
How to do it... 191
Exploiting Elasticsearch 193
How to do it... 194
See also 196
Good old Wireshark 196

Getting ready 196
How to do it... 197
See also 202
This is Sparta 202
Getting ready 203
How to do it... 203
Exploiting Jenkins 208
How to do it... 208
See also 211
Shellver – reverse shell cheatsheet 211
Getting ready 211
How to do it... 211
Generating payloads with MSFvenom Payload Creator (MSFPC) 215
How to do it... 215
Chapter 6: Wireless Attacks - Getting Past Aircrack-ng 220
The good old Aircrack 221
Getting ready 221
How to do it... 221
How it works... 224
Hands-on with Gerix 225
Getting ready 225
How to do it... 225
Dealing with WPAs 230
How to do it... 230
Owning employee accounts with Ghost Phisher 231
How to do it... 232
Pixie dust attack 236
Getting ready 236
How to do it... 237
See also 238
Setting up rogue access points with WiFi-Pumpkin 238
Getting ready 238
How to do it... 239
See also 240
Using Airgeddon for Wi-Fi attacks 240
How to do it... 241
See also 247
Chapter 7: Password Attacks - The Fault in Their Stars 249
Identifying different types of hashes in the wild 249
How to do it... 250
See also 251
Hash-identifier to the rescue 251
How to do it... 251

Cracking with Patator 252
How to do it... 252
Playing with John the Ripper 254
How to do it... 254
See also 255
Johnny Bravo! 256
How to do it... 256
Using ceWL 258
How to do it... 258
Generating wordlists with crunch 259
How to do it... 259
Using Pipal 260
How to do it... 260
Chapter 8: Have Shell, Now What? 263
Spawning a TTY shell 264
How to do it... 264
Looking for weaknesses 267
How to do it... 268
There's more... 270
Horizontal escalation 270
How to do it... 271
Vertical escalation 272
How to do it... 272
Node hopping – pivoting 278
How to do it... 278
There's more... 279
Privilege escalation on Windows 280
How to do it... 280
Pulling a plaintext password with Mimikatz 285
How to do it... 286
Dumping other saved passwords from the machine 288
How to do it... 288
Pivoting 292
How to do it... 292
Backdooring for persistance 292
How to do it... 293
Age of Empire 295
Getting ready 296
How to do it... 296
See also 310
Automating Active Directory (AD) exploitation with DeathStar 310
How to do it... 310
See also 313

Exfiltrating data through Dropbox 313
How to do it... 313
Data exfiltration using CloakifyFactory 315
How to do it... 316

Chapter 9: Buffer Overflows 324
Exploiting stack-based buffer overflows 327
How to do it... 327
Exploiting buffer overflows on real software 334
Getting ready 334
How to do it... 334
SEH bypass 343
How to do it... 345
See also 354
Exploiting egg hunters 354
Getting ready 355
How to do it... 355
See also 359
An overview of ASLR and NX bypass 359
How to do it... 359
See also 361

Chapter 10: Elementary, My Dear Watson - Digital Forensics 362
Using the volatility framework 362
Getting ready 364
How to do it... 364
See also 368
Using Binwalk 368
How to do it... 368
See also 370
Capturing a forensic image with guymager 370
How to do it... 371

Chapter 11: Playing with Software-Defined Radios 373
Radio-frequency scanners 373
Getting ready 374
How to do it... 374
Hands-on with the RTLSDR scanner 375
How to do it... 375
Playing around with gqrx 377
How to do it... 378
See also 380
Kalibrating your device for GSM tapping 381
How to do it... 381
See also 387
Decoding ADS-B messages with Dump1090 387

How to do it... 388
See also 388

Chapter 12: Kali in Your Pocket - NetHunters and Raspberries 389
Installing Kali on Raspberry Pi 389
Getting ready 390
How to do it... 390
Installing NetHunter 391
Getting ready 392
How to do it... 392
Superman typing – human interface device (HID) attacks 396
How to do it... 397
Can I charge my phone? 402
How to do it... 402
Setting up an evil access point 405
How to do it... 405

Chapter 13: Writing Reports 410
Using Dradis 411
How to do it... 411
Using MagicTree 422
How to do it... 422
Using Serpico 425
Getting ready 426
How to do it... 430

Other Books You May Enjoy 437

Index 440

Preface

This book begins with the installation and configuration of Kali Linux to help you perform your tests. You will then learn about methods that will help you gather intel and perform web application exploitation using tools such as Burp. Moving forward, you will also learn how to perform network exploitation by generating payloads using MSFPC, Metasploit, and Cobalt Strike. Next, you will learn about monitoring and cracking wireless networks using Aircrack, Fluxion, and Wifi-Pumpkin. After that, you will learn how to analyze, generate, and crack passwords using tools such as Patator, John the Ripper, and ceWL. Later, you will also learn about some of the tools that help in forensic investigations. Lastly, you will learn how to create an optimum quality pentest report!

By the end of this book, you will know how to conduct advanced and efficient penetration testing activities thanks to the book's crisp and task-oriented recipes.

Who this book is for

This book is aimed at IT security professionals, pentesters, and security analysts who have some basic knowledge of Kali Linux and who want to exploit advanced penetration testing techniques.

What this book covers

`Chapter 1`, *Kali - An Introduction*, explains that while Kali is already pre-equipped with hundreds of amazing tools and utilities to help penetration testers around the globe perform their job efficiently, in this chapter, we will primarily cover some custom tweaks that can be used to facilitate an even better pentesting experience for the users.

`Chapter 2`, *Gathering Intel and Plan Attack Strategies*, dives a little deeper into the content from the previous chapter and looks at a number of different tools available for gathering intel on our target. We start by using the infamous tools of Kali Linux. Gathering information is a very crucial stage of performing a penetration test, as every subsequent step we take after this will be the outcome of all the information we gather during this stage. So it is very important that we gather as much information as possible before jumping into the exploitation stage.

`Chapter 3`, *Vulnerability Assessment – Poking for Holes*, explains that we need to start hunting for vulnerabilities. To become a good pentester, we need to make sure no small details are overlooked.

`Chapter 4`, *Web App Exploitation - Beyond OWASP Top 10*, explains that in the OWASP Top 10, we usually see the most common ways of finding and exploiting vulnerabilities. In this chapter, we will cover some of the uncommon cases you might come across while hunting for bugs in a web application.

`Chapter 5`, *Network Exploitation*, covers some of the uncommon ways in which we can pentest a network and successfully exploit the services we find.

`Chapter 6`, *Wireless Attacks - Getting Past Aircrack-ng*, focuses on different areas of Wi-Fi security from the point of view of monitoring, packet capture, and exporting of data to text files for further processing by third-party tools; from the point of view of attacking, replay attacks, deauthentication, fake access points, and others via packet injection testing. From the point of view of checking, Wi-Fi cards and driver capabilities (capture and injection); and finally, from the point of view of cracking, WEP, and WPA PSK (WPA 1 and 2).

`Chapter 7`, *Password Attacks - the Fault in Their Stars*, explains how a weak password is a well-known scenario where most corporates are compromised. A lot of people use weak passwords that can be brute forced and plaintext can be obtained. In this chapter, we will talk about different ways in which we can crack a password hash obtained during a pentest activity performed on a web app/network, among others.

`Chapter 8`, *Have Shell, Now What?* covers the different ways of escalating our privileges on Linux and Windows systems as well as pivoting to the internal network.

`Chapter 9`, *Buffer Overflows*, introduces the basics of assembly, exploiting buffer overflows, bypassing SEH, egg hunting, and a little bit about ASLR Bypass.

`Chapter 10`, *Elementary, My Dear Watson - Digital Forensics*, explains how memory forensics (sometimes referred to as memory analysis) refers to the analysis of volatile data in a computer's memory dump. It is used to investigate attacks on the system that are stealthy and do not leave data on the hard drive of the computer. In this chapter, we will cover some of the tools that can be used to analyze memory dumps and malicious files, and extract useful information from them.

`Chapter 11`, *Playing with Software-Defined Radios*, explains how the term *software-defined radio* means the implementation of hardware-based radio components, including modulators, demodulators, and tuners, using software. In this chapter, we will cover different recipes and look at multiple ways that RTLSDR can be used to play around with frequencies and the data being transported through it.

`Chapter 12`, *Kali in Your Pocket - NetHunters and Raspberries*, talks about setting up Kali Linux on Raspberry Pi and compatible cell phones and using it to perform a number of cool attacks on the network.

`Chapter 13`, *Writing Reports*, goes through one of the most important steps of a pentesting project – the report. A good report must contain every detail of the vulnerability. Our agenda is to keep it as detailed as possible, which may help the right person in the department understand all the details and work around it with a perfect patch. There are different ways to create a pentesting report. In this chapter, you will learn a few tools that we can use to generate a good report that covers everything in detail.

To get the most out of this book

The OS required is Kali Linux, with at least 2 GB of RAM recommended and 20-40 GB of hard disk space. The hardware required for the device would be an RTLSDR device for `Chapter 11`, *Playing with Software-Defined Radios*, and any of the devices mentioned in the following link for `Chapter 12`, *Kali in Your Pocket – NetHunters and Raspberries*: `https://www.offensive-security.com/kali-linux-nethunter-download/`

You will also require an Alfa card for `Chapter 6`, *Wireless Attacks – Getting Past Aircrack-ng*.

Download the color images

We also provide a PDF file that has color images of the screenshots/diagrams used in this book. You can download it here: `https://www.packtpub.com/sites/default/files/downloads/9781789952308_ColorImages.pdf`.

Conventions used

There are a number of text conventions used throughout this book.

`CodeInText`: Indicates code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles. Here is an example: "Choose the `xfce-session` option (in our case, 3) and press *Enter*."

Any command-line input or output is written as follows:

```
update-alternatives --config x-session-manager
```

Bold: Indicates a new term, an important word, or words that you see on screen. For example, words in menus or dialog boxes appear in the text like this. Here is an example: "In the **Payloads** tab, we select the **Payload type** as **Extension-generated**."

Warnings or important notes appear like this.

Tips and tricks appear like this.

Sections

In this book, you will find several headings that appear frequently (*Getting ready*, *How to do it...*, *How it works...*, *There's more...*, and *See also*).

To give clear instructions on how to complete a recipe, use these sections as follows:

Getting ready

This section tells you what to expect in the recipe and describes how to set up any software or any preliminary settings required for the recipe.

How to do it…

This section contains the steps required to follow the recipe.

How it works…

This section usually consists of a detailed explanation of what happened in the previous section.

There's more…

This section consists of additional information relating to the recipe in order to make you more knowledgeable about the recipe.

See also

This section provides helpful links to other useful information for the recipe.

Get in touch

Feedback from our readers is always welcome.

General feedback: If you have questions about any aspect of this book, mention the book title in the subject of your message and email us at `customercare@packtpub.com`.

Errata: Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you have found a mistake in this book, we would be grateful if you would report this to us. Please visit `www.packt.com/submit-errata`, selecting your book, clicking on the Errata Submission Form link, and entering the details.

Piracy: If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at `copyright@packt.com` with a link to the material.

If you are interested in becoming an author: If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit `authors.packtpub.com`.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions, we at Packt can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about Packt, please visit `packt.com`.

Disclaimer

The information within this book is intended to be used only in an ethical manner. Do not use any information from the book if you do not have written permission from the owner of the equipment. If you perform illegal actions, you are likely to be arrested and prosecuted to the full extent of the law. Packt Publishing does not take any responsibility if you misuse any of the information contained within the book. The information herein must only be used while testing environments with proper written authorizations from appropriate persons responsible.

1
Kali - An Introduction

Kali was first introduced in 2012 with a completely new architecture. This Debian-based distribution was released with over 300 specialized tools for penetration testing and digital forensics. It is maintained and funded by Offensive Security Ltd, and the core developers are Mati Aharoni, Devon Kearns, and Raphaël Hertzog.

Kali 3.0 came into the picture in 2018 with tons of new updates, bug fixes such as AMD Secure Memory Encryption Support, and increased memory limits.

In the previous edition of this book, we saw some of the great tools in Kali that help penetration testers around the globe to perform their job efficiently. In this chapter, we will primarily cover the installation of Kali and setting up different desktop environments, as well as some custom tools that will help us.

In this chapter, we will cover the following recipes:

- Configuring Kali Linux
- Configuring the Xfce environment
- Configuring the MATE environment
- Configuring the LXDE environment
- Configuring the E17 environment
- Configuring the KDE environment
- Prepping Kali with custom tools
- Zone Walking using DNSRecon
- Setting up I2P for anonymity
- Pentesting VPN's ike-scan
- Setting up proxychains
- Going on a hunt with Routerhunter

Configuring Kali Linux

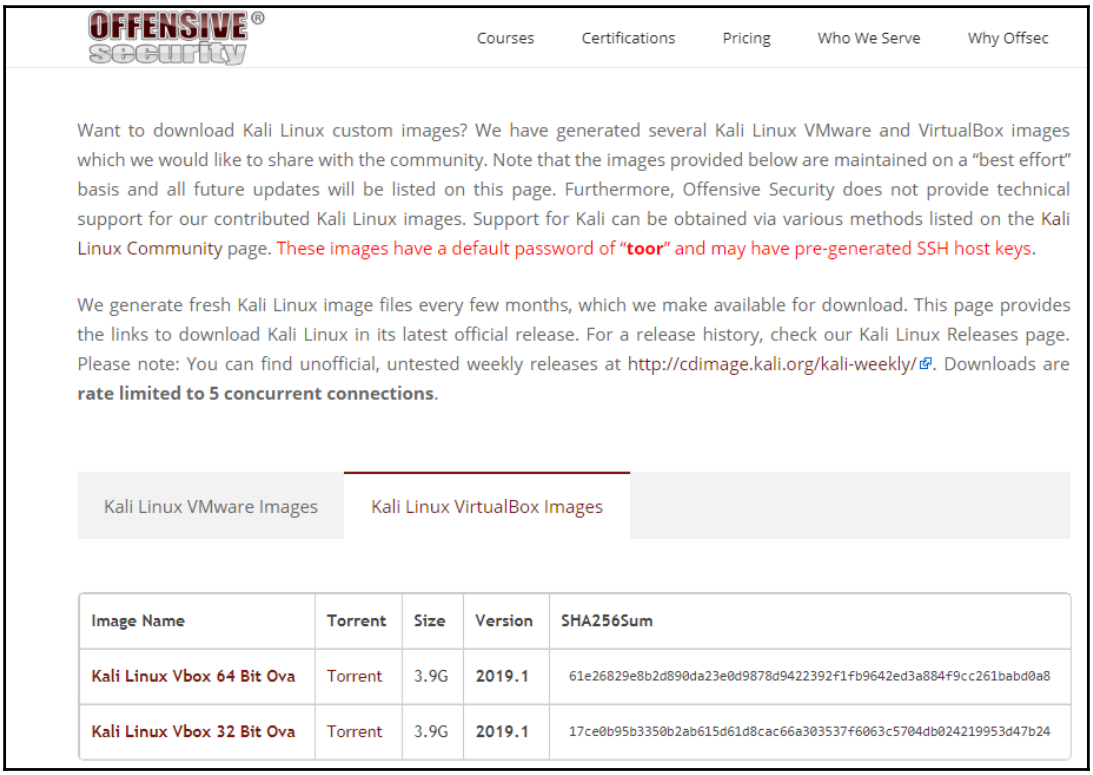
We will use the official Kali Linux official ISO provided by Offensive Security to install and configure different desktop environments.

Getting ready

To start with this recipe, we will use the 64-bit Kali Linux ISO listed on the Offensive Security website: `https://www.kali.org/downloads/`.

For users looking to configure Kali for a virtual machine such as VMware and VirtualBox, a prebuilt image of the Linux can be downloaded from the following URL: `https://www.offensive-security.com/kali-linux-vmware-virtualbox-image-download/`.

We will use the virtual image in this chapter and customize it with some additional tools. We can download it from the website, as shown in the following screenshot:



OFFENSIVE security

Courses Certifications Pricing Who We Serve Why Offsec

Want to download Kali Linux custom images? We have generated several Kali Linux VMware and VirtualBox images which we would like to share with the community. Note that the images provided below are maintained on a "best effort" basis and all future updates will be listed on this page. Furthermore, Offensive Security does not provide technical support for our contributed Kali Linux images. Support for Kali can be obtained via various methods listed on the Kali Linux Community page. These images have a default password of "**toor**" and may have pre-generated SSH host keys.

We generate fresh Kali Linux image files every few months, which we make available for download. This page provides the links to download Kali Linux in its latest official release. For a release history, check our Kali Linux Releases page. Please note: You can find unofficial, untested weekly releases at http://cdimage.kali.org/kali-weekly/. Downloads are **rate limited to 5 concurrent connections**.

Kali Linux VMware Images | Kali Linux VirtualBox Images

Image Name	Torrent	Size	Version	SHA256Sum
Kali Linux Vbox 64 Bit Ova	Torrent	3.9G	2019.1	61e26829e8b2d890da23e0d9878d9422392f1fb9642ed3a884f9cc261babd0a8
Kali Linux Vbox 32 Bit Ova	Torrent	3.9G	2019.1	17ce0b95b3350b2ab615d61d8cac66a303537f6063c5704db024219953d47b24

How to do it...

1. Double-click the VirtualBox image; it should open with VirtualBox.
2. Click **Import**.
3. Start the machine and enter the password `toor`.
4. Now, Kali is by default configured with Gnome Desktop Environment.

How it works...

With the prebuilt image, you don't need to worry about the installation process. You can consider it as a ready-to-go solution. Simply click on **Run** and the virtual machine will boot up the Linux just like a normal machine.

Configuring the Xfce environment

Xfce is a free, fast, and lightweight desktop environment for Unix and Unix-like platforms. It was started by Olivier Fourdan in 1996. The name **Xfce** originally stood for **XForms Common Environment**, but since that time Xfce has been rewritten twice and no longer uses the XForms toolkit.

How to do it...

1. We start by using the following command to install Xfce, along with all its plugins and goodies. If for some reason it fails, we should run `apt update` first:

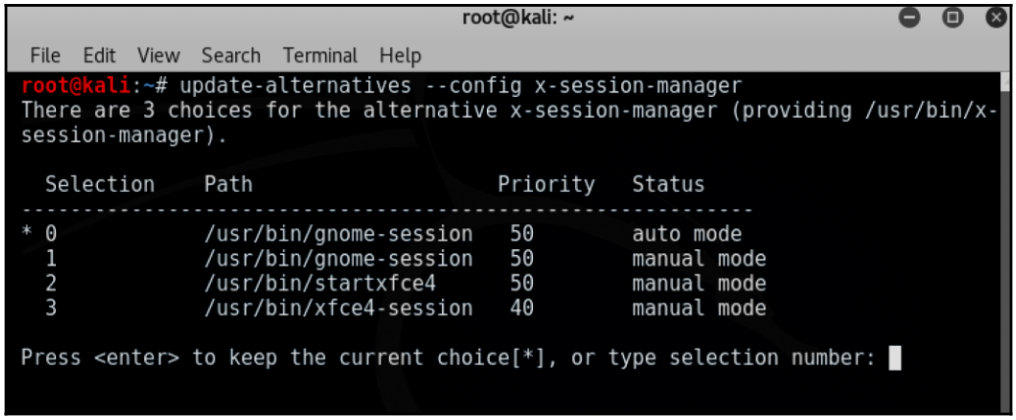
   ```
   apt-get install kali-defaults kali-root-login desktop-base xfce4
   xfce4-places-plugin xfce4-goodies
   ```

   The following screenshot shows the preceding command:

2. Type `Y` when it asks for confirmation on additional space requirements.
3. Select **OK** on the dialog box that appears.
4. Select **Lightdm** as our default desktop manager and press *Enter*.
5. When the installation is complete, open a Terminal window and type the following command:

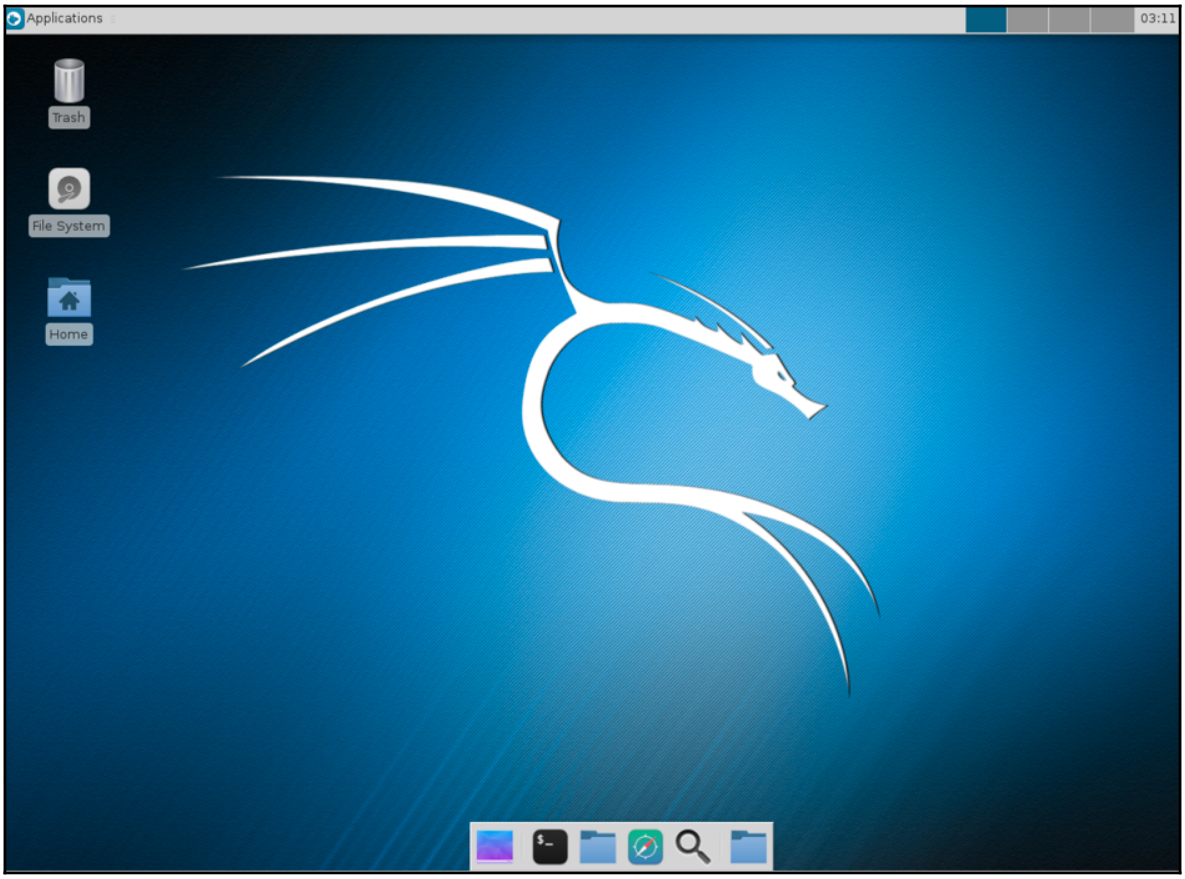
```
update-alternatives --config x-session-manager
```

The following screenshot shows the output of the preceding command:



6. Choose the `xfce-session` option (in our case, **3**) and press *Enter*.

7. Log out and log in again, and we will see the Xfce environment:



Now let's have a look at the configuration of MATE environment.

Configuring the MATE environment

The MATE desktop environment is the continuation of GNOME 2. It provides an intuitive and attractive desktop environment using traditional metaphors for Linux and other Unix-like operating systems. The latest version of MATE (1.20) was released on 07-02-2018, which added a lot of fixes and upgraded the theme.

The complete list of features can be viewed here: `https://mate-desktop.org/blog/2018-02-07-mate-1-20-released/`.

In this recipe, we will learn how to install MATE on Kali Linux.

How to do it...

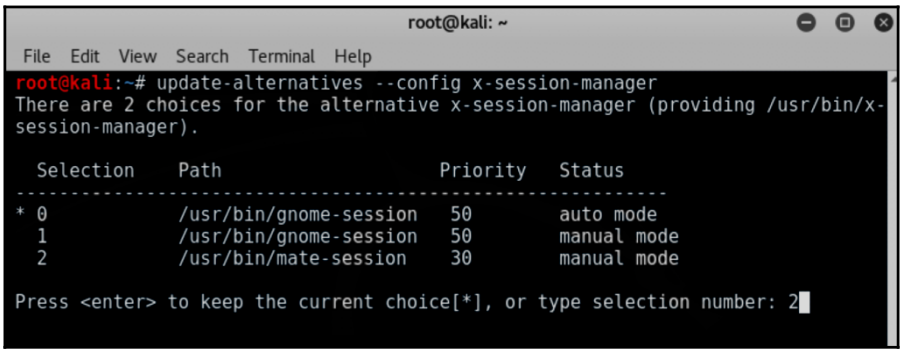
1. We start by using the following command to install the MATE environment:

   ```
   apt-get install desktop-base mate-desktop-environment
   ```

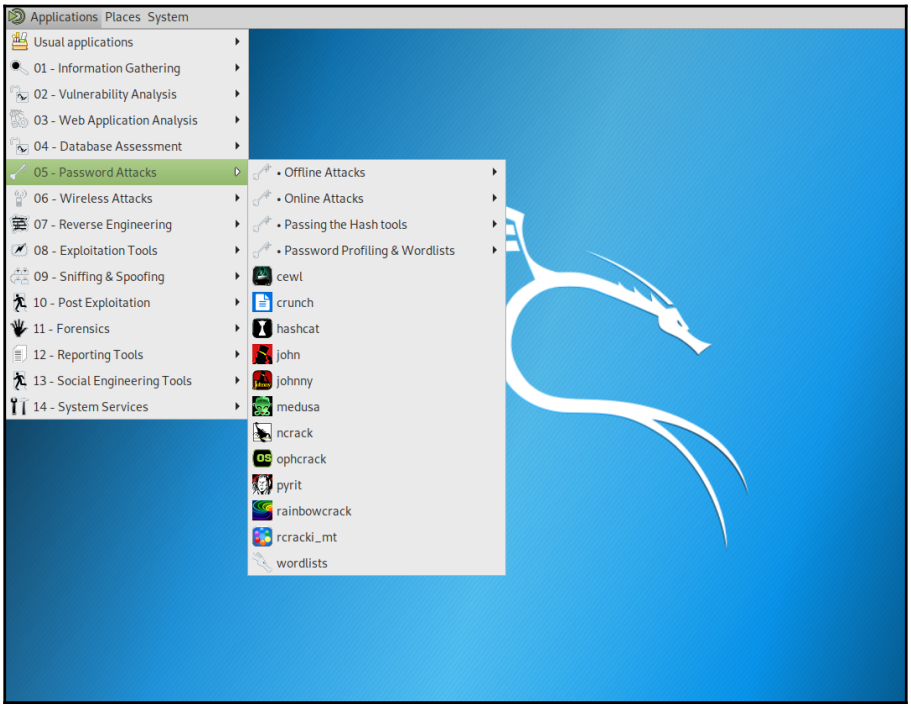
2. Type `Y` when it asks for confirmation on additional space requirements.
3. When installation is complete, we will use the following command to set MATE as our default environment:

   ```
   update-alternatives --config x-session-manager
   ```

4. Choose the `mate-session` option (in our case, **2**) and press *Enter*:



5. Log out and log in again, and we will see the MATE environment:



Now let's have a look at the configuration of LXDE environment.

Configuring the LXDE environment

LXDE is a free open source environment written in C using the GTK+ toolkit for Unix and other POSIX platforms. **LXDE** stands for **Lightweight X11 Desktop Environment**.

LXDE is the default environment for many operating systems, such as Knoppix, Raspbian, and Lubuntu.

How to do it...

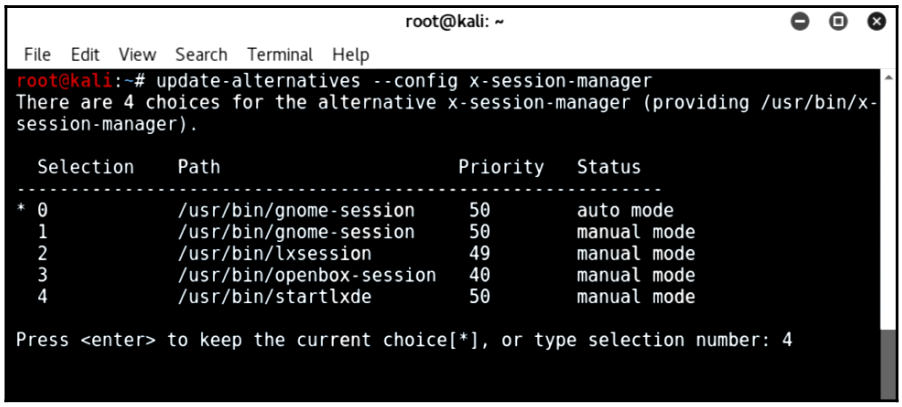
1. We start by using the following command to install LXDE:

```
apt-get install lxde-core lxde
```

2. Type `Y` when it asks for confirmation on additional space requirements.
3. When the installation is complete, open a Terminal window and type the following command:

```
update-alternatives --config x-session-manager
```

The following screenshot shows the output of the preceding command:



4. Choose the `startlxde` option session (in our case, **4**) and press *Enter*.
5. Log out and log in again, and we will see the LXDE environment:

Now let's have a look at the configuration of E17 environment.

Configuring the E17 environment

Enlightenment, otherwise known as **E**, is a window manager for the X Windows system. It was first released in 1997. It has lots of features, such as engage, virtual desktop, and tiling.

How to do it...

1. Due to compatibility issues and hassle regarding dependencies, it is better to download Kali with the E17 environment directly from the following URL: `https://www.kali.org/downloads/`.
2. The steps to set it up are simple: we just have to double-click and start the VM in VirtualBox or VMware.

Configuring the KDE environment

K Desktop Environment (**KDE**) is an open source graphical desktop environment for UNIX workstations. It was initially called Kool Desktop Environment. Matthias Ettrich first launched the KDE project in 1996 with the goal of making the UNIX platform more attractive and easy to use. In this recipe, we will learn how to set up KDE on Kali.

How to do it...

1. We use the following command to install KDE:

```
apt-get install kali-defaults kali-root-login desktop-base kde-
plasma-desktop
```

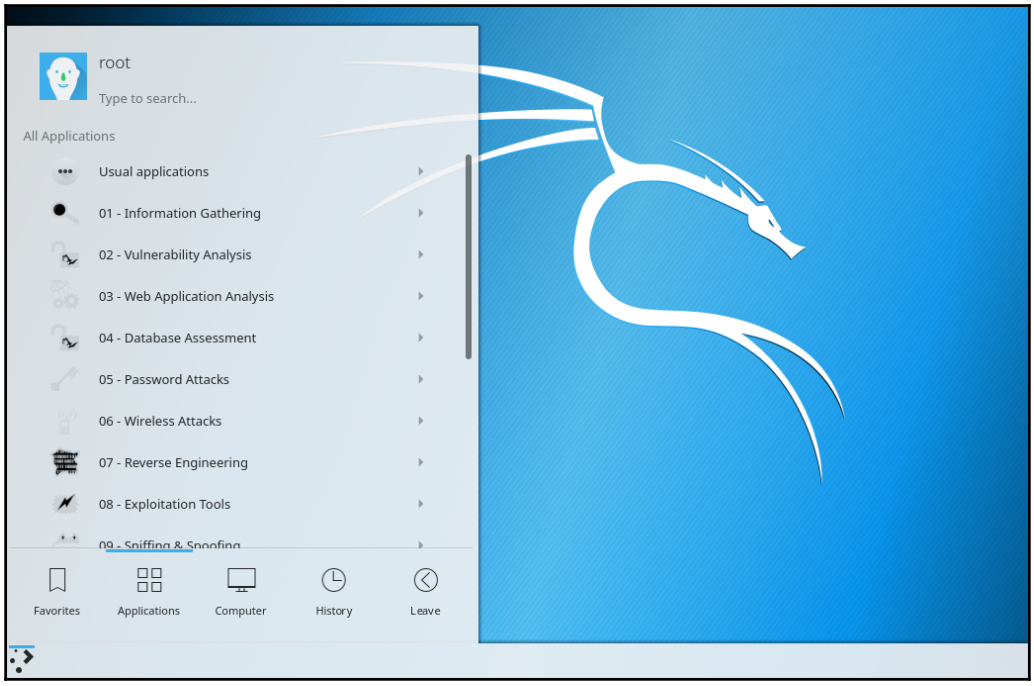
2. Type `Y` when it asks for confirmation on additional space requirements.
3. Click **OK** on both the windows that pop up.
4. When the installation is complete, we open a Terminal window and type the following command:

```
update-alternatives --config x-session-manager
```

The following screenshot shows the output of the preceding command:

```
File  Edit  View  Search  Terminal  Help
root@kali:~# update-alternatives --config x-session-manager
There are 2 choices for the alternative x-session-manager (providing /usr/bin/x-session-manager).

  Selection    Path                    Priority   Status
------------------------------------------------------------
* 0            /usr/bin/gnome-session   50        auto mode
  1            /usr/bin/gnome-session   50        manual mode
  2            /usr/bin/startkde        40        manual mode

Press <enter> to keep the current choice[*], or type selection number: 2
update-alternatives: using /usr/bin/startkde to provide /usr/bin/x-session-manager (x-session-manager) in manual mode
root@kali:~#
```

5. Choose the `startkde` option (in our case, **2**) and press *Enter*.
6. Log out and log in again, and we will see the KDE environment:



Kali has already provided prebuilt images of different desktop environments. These can be downloaded from `https://www.kali.org/downloads/`.

Prepping with custom tools

In this recipe, we will set up a few tools beforehand; not to worry, we will be covering their usage in detail in later chapters.

Getting ready

Here is a list of some tools that we will need before we dive deeper into penetration testing. Don't worry, we will learn about their usage with some real-life examples in the next few chapters. But those of us who are excited about them right now can run the following simple commands to view the `-help` section where `toolname` is the name of the tool we would like to view the help of:

```
toolname -help
toolname -h
```

How to do it...

We will be looking at two tools in this section.

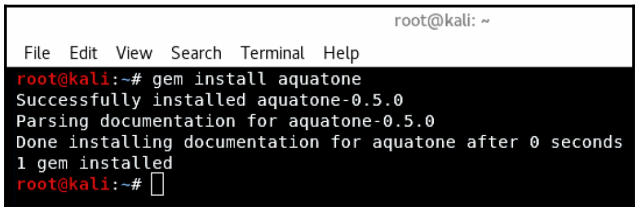
Aquatone

Aquatone is a tool for visually inspecting websites across a large amount of hosts and is convenient for quickly gaining an overview of an HTTP-based attack surface. Aquatone has four major modules: discover, scanner, gather, and takeover. Each of these can be used to perform in-depth enumeration of a target:

1. We will use a simple command to install `aquatone`:

   ```
   gem install aquatone
   ```

   The following screenshot shows the output of the preceding command:



2. Next, we create a directory in `/root/folder` using the following command:

```
mkdir /root/aquatone/
```

3. As aquatone uses different modules to hunt for subdomains, we will have to configure aquatone's discovery module before running it.
4. For example, to configure the `shodan`, we can use the following command:

```
aquatone-discover --set-key shodan XXXXXXXXXXX
```

The following screenshot shows the output of the preceding command:

```
root@kali:~# aquatone-discover --set-key shodan
aM
Saved key shodan with value IeREX9s
```

5. Similarly, we can set keys for other services too, such as Censys and PassiveTotal.
6. Once it is all set, we can start our subdomain hunting. We can do this using the following command:

```
aquatone-discover -d domain.com
```

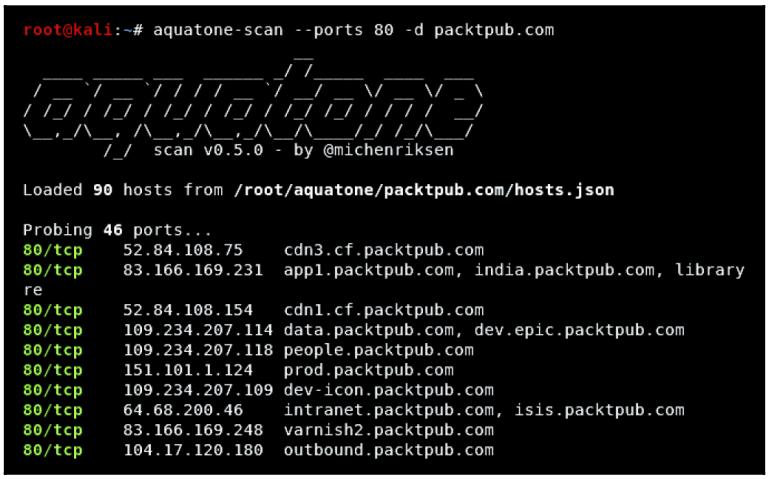
The following screenshot shows the output of the preceding command:

```
root@kali:~# aquatone-discover -d packtpub.com -t 150
                                        __
  ____ _____ ___  ______ _/ /_____  ____  ___
 / __ `/ __ `/ / / / __ `/  __/  __ \/ __ \/ _ \
/ /_/ / /_/ / /_/ / /_/ / /_/ /_/ / / / / /  __/
\__,_/\__, /\__,_/\__,_/\__/\____/_/ /_/\___/
        /_/  discover v0.5.0 - by @michenriksen

Identifying nameservers for packtpub.com... Done
Using nameservers:

 - 64.68.196.10
 - 64.68.192.10
 - 198.41.222.254
 - 64.68.197.10

Checking for wildcard DNS... Done

Running collector: VirusTotal... Skipped
 -> Key 'virustotal' has not been set
Running collector: Google Transparency Report...
```

7. Aquatone also allows us to set a custom wordlist by using the `-w` flag, and we can also set the threads by using the `-t` flag.
8. By default, aquatone stores the output in TXT as well as JSON format in the `/root/aquatone/` directory.
9. After we find the subdomains, we can use the aquatone scanner to scan for open ports on the discovered hosts. Let's look at an example:

```
aquatone-scan --ports 80 -d packtpub.com
```

The following screenshot shows the output of the preceding command:



```
root@kali:~# aquatone-scan --ports 80 -d packtpub.com
                                                    __
  ____ _____ ___  ______ _/ /_____  ____  ___
 / __ `/ __ `/ / / / __ `/ __/ __ \/ __ \/ _ \
/ /_/ / /_/ / /_/ / /_/ / /_/ /_/ / / / /  __/
\__,_/\__, /\__,_/\__,_/\__/\____/_/ /_/\___/
        /_/  scan v0.5.0 - by @michenriksen

Loaded 90 hosts from /root/aquatone/packtpub.com/hosts.json

Probing 46 ports...
80/tcp    52.84.108.75    cdn3.cf.packtpub.com
80/tcp    83.166.169.231  app1.packtpub.com, india.packtpub.com, library
re
80/tcp    52.84.108.154   cdn1.cf.packtpub.com
80/tcp    109.234.207.114 data.packtpub.com, dev.epic.packtpub.com
80/tcp    109.234.207.118 people.packtpub.com
80/tcp    151.101.1.124   prod.packtpub.com
80/tcp    109.234.207.109 dev-icon.packtpub.com
80/tcp    64.68.200.46    intranet.packtpub.com, isis.packtpub.com
80/tcp    83.166.169.248  varnish2.packtpub.com
80/tcp    104.17.120.180  outbound.packtpub.com
```

10. This will look for the domain's `hosts.json` file in the `aquatone` directory. Aquatone by default has four inbuilt port scanning flags (small, medium, large, and huge). These flags will decide the number of ports being scanned on the hosts, or we can define custom ports by using the `-ports` flag.
    - `aquatone-gather`: This tool makes a connection to the web services found using the discover and scanner modules of aquatone and takes screenshots of discovered web pages for later analysis.
    - `aquatone-takeover`: This module is used to find subdomains that are vulnerable to the subdomain takeover vulnerability.

Let's refer to the following screenshot:

```
root@kali:~# aquatone-takeover -d packtpub.com
                                  __
  ____ _____ ___  ______ _/ /_____  ____  ___
 / __ `/ __ `/ / / / __ `/ __/ __ \/ __ \/ _ \
/ /_/ / /_/ / /_/ / /_/ / /_/ /_/ / / / /  __/
\__,_/\__, /\__,_/\__,_/\__/\____/_/ /_/\___/
        /_/  takeover v0.5.0 - by @michenriksen

Loaded 90 hosts from /root/aquatone/packtpub.com/hosts.json
Loaded 25 domain takeover detectors

Identifying nameservers for packtpub.com... Done
Using nameservers:

 - 64.68.197.10
 - 198.41.222.254
 - 64.68.192.10
 - 64.68.196.10

Checking hosts for domain takeover vulnerabilities...
```

Subfinder

Subfinder is considered as a successor to sublist3r. It is amazingly fast and finds valid subdomains using passive online sources such as Ask, Archive.is, Baidu, Bing, Censys, CertDB, CertSpotter, Commoncrawl, CrtSH, DnsDB and so on.

1. Install subfinder. It needs Go to be installed, which we can install by using the following command:

   ```
   apt install golang
   ```

   The following screenshot shows the output of the preceding command:

```
root@kali: ~                    root@kali: ~/subfinder
root@kali:~/subfinder# apt install golang
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  golang-1.10 golang-1.10-doc golang-1.10-go golang-1.10-src golang-doc
  golang-go golang-src pkg-config
Suggested packages:
  bzr | brz
The following NEW packages will be installed:
  golang golang-1.10 golang-1.10-doc golang-1.10-go golang-1.10-src golang-doc
  golang-go golang-src pkg-config
0 upgraded, 9 newly installed, 0 to remove and 1543 not upgraded.
Need to get 42.7 MB of archives.
After this operation, 227 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
0% [Connecting to http.kali.org]
```

2. Next, we clone `subfinder` by using the following command:

```
git clone https://github.com/subfinder/subfinder.git
```

The following screenshot shows the output of the preceding command:

```
root@kali: ~
File  Edit  View  Search  Terminal  Tabs  Help
root@kali: ~                    root@kali: ~
root@kali:~# git clone https://github.com/subfinder/subfinder.git
Cloning into 'subfinder'...
remote: Enumerating objects: 2583, done.
remote: Total 2583 (delta 0), reused 0 (delta 0), pack-reused 2583
Receiving objects: 100% (2583/2583), 497.66 KiB | 404.00 KiB/s, done.
Resolving deltas: 100% (1316/1316), done.
root@kali:~#
```

Or you can download and save it from `https://github.com/subfinder/subfinder`.

3. To install subfinder, we go to the cloned directory and run the `go build` command.
4. Once the installation is complete, we will need a wordlist for it to run, so we can download dnspop's list. This list can be used in the previous recipe too: `https://github.com/bitquark/dnspop/tree/master/results`.
5. Now that both are set up, we browse into subfinder's directory and run it using the `./subfinder -h` command.

The following screenshot shows the output of the preceding command:

```
root@kali:~/subfinder# ./subfinder -h
Usage of ./subfinder:
  -b    Use bruteforcing to find subdomains
  -d string
        Domain to find subdomains for
  -dL string
        List of domains to find subdomains for
  -exclude-sources string
        List of sources to exclude from enumeration
  -nW
        Remove Wildcard Subdomains from output
  -no-color
        Don't Use colors in output (default true)
  -no-passive
        Do not perform passive subdomain enumeration
  -o string
        Name of the output file (optional)
  -oD string
        Directory to output results to
  -oJ
        Write output in JSON Format
```

6. To run it against a domain with our wordlist, we use the following command:

   `./subfinder -w /path/to/wordlist -d hostname.com`

   If we do not specify a wordlist the tool will run with a default wordlist as shown in the following screenshot:

```
}root@kali:~/subfinder# ./subfinder -d packtpub.com -t 20
================================================
-=Subfinder v1.1.3 github.com/subfinder/subfinder
================================================


Running Source: Ask
Running Source: Archive.is
Running Source: Baidu
Running Source: Bing
Running Source: CertDB
Running Source: CertificateTransparency
Running Source: Certspotter
Running Source: Commoncrawl
Running Source: Crt.sh
Running Source: Dnsdb
Running Source: DNSDumpster
Running Source: DNSTable
Running Source: Dogpile
```

Once the enumeration is complete, the output will be shown onscreen as follows:

```
Total 75 Unique subdomains found for packtpub.com

%2Fwww.packtpub.com
3www.packtpub.com
Www.packtpub.com
account.packtpub.com
api-dev.packtpub.com
app.packtpub.com
app1.packtpub.com
applications.packtpub.com
auth-api.packtpub.com
authorportal.packtpub.com
authors.packtpub.com
birmingham.packtpub.com
careers.packtpub.com
cdn1.cf.packtpub.com
cdn1.packtpub.com
cdn2.cf.packtpub.com
cdn2.packtpub.com
cdn3.cf.packtpub.com
```

7. Subfinder is also designed to work with services such as `shodan`, `censys`, and `virustotal`, but they need to be configured in the `config.json` file shown here:

```
root@kali:~/subfinder# cat config.json
{
        "virustotalApikey": "",
        "passivetotalUsername": "",
        "passivetotalKey": "",
        "securitytrailsKey": "",
        "riddlerEmail": "",
        "riddlerPassword": "",
        "censysUsername": "",
        "censysSecret": "",
        "shodanApiKey": "
```

There's more...

A subdomain takeover vulnerability exists when a service that previously pointed to a subdomain is removed but the CNAME record still exists. More information can be read about it at the following GitHub link: `https://github.com/EdOverflow/can-i-take-over-xyz/`.

Aquatone-takeover is based on the same methodology described by EdOverflow at the preceding URL.

Zone Walking using DNSRecon

Zone Walking is a technique that is used by attackers to enumerate the full content of DNSSEC-signed DNS zones. We will cover more about it in later chapters; in this recipe, we will use DNSRecon.

Getting ready

DNSRecon is already included in Kali Linux, and we can use it for Zone Walking. Zone Walking is a technique used to find subdomains using domains whose NSEC records are set. However, before we jump into Zone Walking, let's take a quick look at the other features of this tool.

How to do it...

1. To view the help, we type the following:

   ```
   dnsrecon -h
   ```

   The following screenshot shows the output of the preceding command:

```
root@kali:~# dnsrecon -h
usage: dnsrecon.py [-h] [-d DOMAIN] [-n NS_SERVER] [-r RANGE] [-D DICTIONARY]
                   [-f] [-t TYPE] [-a] [-s] [-g] [-b] [-k] [-w] [-z]
                   [--threads THREADS] [--lifetime LIFETIME] [--db DB]
                   [-x XML] [-c CSV] [-j JSON] [--iw] [-v]

optional arguments:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Target domain.
  -n NS_SERVER, --name_server NS_SERVER
                        Domain server to use. If none is given, the SOA of the
                        target will be used.
  -r RANGE, --range RANGE
                        IP range for reverse lookup brute force in formats
                        (first-last) or in (range/bitmask).
  -D DICTIONARY, --dictionary DICTIONARY
                        Dictionary file of subdomain and hostnames to use for
                        brute force. Filter out of brute force domain lookup,
                        records that resolve to the wildcard defined IP
```

2. To do a simple recon of name servers, A records, SOA records, MX records, and so on, we can run the following command:

```
dnsrecon -d packtpub.com -n 8.8.8.8
```

The following screenshot shows the output of the preceding command:

```
root@kali:~# dnsrecon -d packtpub.com -n 8.8.8.8
[*] Performing General Enumeration of Domain: packtpub.com
[-] DNSSEC is not configured for packtpub.com
[*]      SOA dns1.easydns.com 64.68.192.10
[*]      NS dns3.easydns.org 64.68.196.10
[*]      Bind Version for 64.68.196.10 lon3
[*]      NS dns3.easydns.org 2620:49:3::10
[*]      NS dns4.easydns.info 64.68.197.10
[*]      Bind Version for 64.68.197.10 nyc2
[*]      NS dns4.easydns.info 2620:49:4::10
[*]      NS dns2.easydns.net 198.41.222.254
[*]      Bind Version for 198.41.222.254 Salt-master
[*]      NS dns2.easydns.net 2400:cb00:2049:1::c629:defe
[*]      NS dns1.easydns.com 64.68.192.10
[*]      Bind Version for 64.68.192.10 Salt-master
[*]      NS dns1.easydns.com 2400:cb00:2049:1::a29f:1835
[*]      MX packtpub-com.mail.protection.outlook.com 104.47.21.36
[*]      MX packtpub-com.mail.protection.outlook.com 104.47.20.36
[*]      A packtpub.com 83.166.169.231
[*]      TXT packtpub.com v=spf1 ip4:109.234.197.32/27 ip4:83.166.169.224/27 ip4:109.234.207.96/27
ip4:168.245.75.197 a:zgateway.zuora.com include:spf1.mailgun.org include:spf2.mailgun.org include:s
pf.protection.outlook.com include:servers.mcsv.net include:spf.mandrillapp.com include:sendgrid.net
 include:amazonses.com -all
[*]      TXT packtpub.com google-site-verification=aYn6H9fduNTMAWna17iNt1G1VNPxaakxn2Vta5qgok0
[*]      TXT packtpub.com _globalsign-domain-verification=6RYPIPUO2OQDU0pqaDmaEeWmISV7Tz3QQBCYgOv3b
r
```

3. Now let's take an example of a domain that has NSEC records. To do a zone walk, we can simply run the following command:

```
dnsrecon -z -d icann.org -n 8.8.8.8
```

The following screenshot shows the output of the preceding command:

```
root@kali:~# dnsrecon -z -d icann.org -n 8.8.8.8
[*] Performing General Enumeration of Domain: icann.org
[*] DNSSEC is configured for icann.org
[*] DNSKEYs:
[*]     NSEC3 ZSK RSASHA1NSEC3SHA1 03010001afeb7eb6eff618ee75d06f2e eeb109b1
d 49f756f08a3a1fc1c891b6d9b07972d5 e6724971b19f77dc97a146db770b2796 4391f8fe
2 a1e0178ee01b25153c59fb44619b63e0 2a6d9a0ec1413227a6c80db97f882b29 5e559ba0
0 09d14ead
[*]     NSEC3 ZSK RSASHA1NSEC3SHA1 03010001dec9d1b7cde251f023c85673 7dbb7b36
4 7c99e59e7a814a3db2c4b078f1507486 aa926b27f2212bd66f0256d04341c7cf 4a74d63d
3 8f2028e5db6b1142b2710c1c2dea74aa 2bc82b3502e320e0623ae76409401866 bd6a2eb5
d f57e757b
[*]     NSEC3 KSk RSASHA1NSEC3SHA1 03010001ac4470a63a03ae738fe2ce3f 3eb540c8
a c01f6219e778bad4374d8e1ee1e4b86e 8c68d547bf97b0bf83cd0261250512ac 88a568db
c 5ace22c2bdba20ff927d0c8735a620ce a79064cb99766285f6e40c9021b9b5b7 89b188c0
5 bec905cbccae5bfa80bc52694156265a 47637cf81cb5b83b524d3fa13d60945f 1ec16cad
9 fdcee3d845a4f520053a9d841e455023 caa12796593bc9b853e0989ce32c9421 a109687c
2 6091fe9767e3b65e1b45017461d3da5e a0868aa41d2576b8fad36a9a5d159a86 2450aaaf
[*]     NSEC3 KSk RSASHA1NSEC3SHA1 03010001d2aaf913851635511877ae00 c0b5be12
2 1c0c403dd0dca76d3d3c70178cf48b3b 05df3c2855822da6a7e2670e294e6d37 e650e6b1
4 1622846f83c46c6051db00d019ff8a6e d3d19bc3f4147ba6fa6a808b5d3283c1 d0c15e6f
d a02d9f4d7f7a812f7a287490c20ee3bd 5d6825c30f988b19a855fa9a842392f9 bac656bc
1 5195a11188b1741d38a1e06d9294c692 05c662c35bc50c502cdff440565e2662 48cf0fd4
e 7c74ef4fe0faf589a874b12643fdf925 5600f934303989655edb73003652c3a0 f33f8f8b
[*]      SOA sns.dns.icann.org 192.0.32.162
[*]      NS ns.icann.org 199.4.138.53
[*]      Bind Version for 199.4.138.53 NSD 4.1.15
[*]      NS ns.icann.org 2001:500:89::53
[*]      NS a.iana-servers.net 199.43.135.53
```

4. We can do this manually by using the `dig` command along with `dig +short NSEC domainname.com`.
5. The previous `dig` command will throw us one subdomain, and then we can rerun the same command with the subdomain we got in previous step to find the next subdomain: `dig +short NSEC a.domain.com`.

There's more...

When signing a zone, DNSSEC automatically chains all labels in alphabetical order using NSEC Resource Records. This is used to prove the absence of names.

For example, if someone requests the non-existent name name3, the name server responds with the NSEC entry name2 NSEC name5, indicating that no other entry exists between name2 and name5. We take advantage of that by starting with the first entry and then getting all domains by calling successive queries and getting other subdomains.

Setting up I2P for anonymity

Invisible Internet Project (**I2P**) is an unknown network layer. It offers P2P communication. To set up an anonymous connection, the user's traffic is encrypted (end to end) and is sent through a network of roughly 55,000 computers, which is distributed around the world and owned by volunteers.

How to do it...

1. To install I2P, we need to first check whether `apt-transport-https` and `curl` are installed:

   ```
   sudo apt-get install apt-transport-https curl
   ```

2. Now we can install the tool using the following command:

   ```
   apt install i2p
   ```

   The following screenshot shows the output of the preceding command:

```
root@kali:~# apt install i2p
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  famfamfam-flag-png i2p-router libeclipse-jdt-core-java libel-api-java libgetopt-java
  libjbigi-jni libjetty9-java libjsp-api-java libservice-wrapper-java libservice-wrapper-jni
  libservlet3.1-java libtaglibs-standard-impl-java libtaglibs-standard-jstlel-java
  libtaglibs-standard-spec-java libtomcat9-java libwebsocket-api-java service-wrapper ttf-dejavu
  ttf-dejavu-core ttf-dejavu-extra
Suggested packages:
  privoxy syndie libgetopt-java-doc jetty9 libservice-wrapper-doc tomcat9
The following NEW packages will be installed:
  famfamfam-flag-png i2p i2p-router libeclipse-jdt-core-java libel-api-java libgetopt-java
  libjbigi-jni libjetty9-java libjsp-api-java libservice-wrapper-java libservice-wrapper-jni
  libservlet3.1-java libtaglibs-standard-impl-java libtaglibs-standard-jstlel-java
  libtaglibs-standard-spec-java libtomcat9-java libwebsocket-api-java service-wrapper ttf-dejavu
  ttf-dejavu-core ttf-dejavu-extra
0 upgraded, 21 newly installed, 0 to remove and 1543 not upgraded.
Need to get 25.1 MB of archives.
After this operation, 33.2 MB of additional disk space will be used.
```

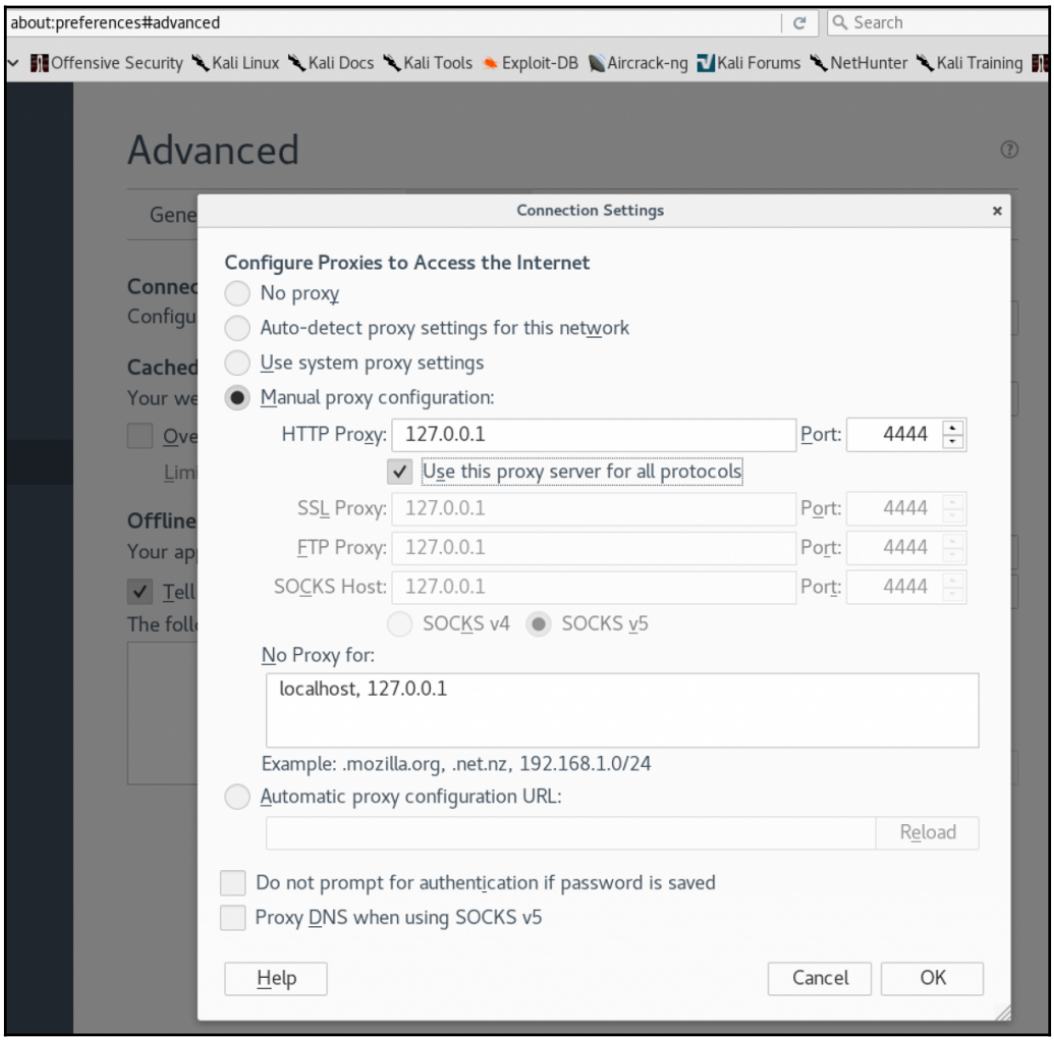
3. When the installation is complete, we can run the service by using the following command:

   ```
   i2prouter start
   ```

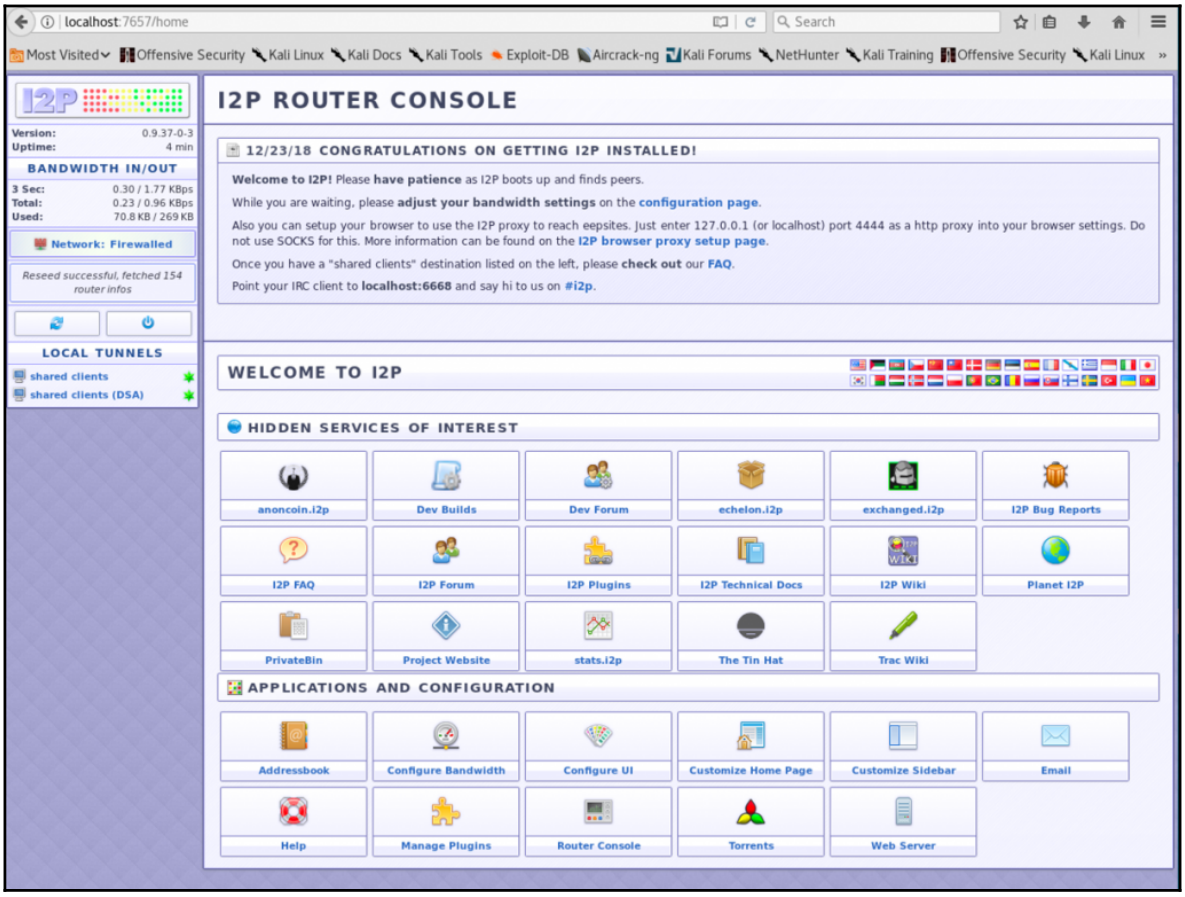
4. We should not run it as root so we log in as another account and run the command as shown in the following screenshot:

```
root@kali:~# su test
test@kali:/root$ i2prouter start
Starting I2P Service...
Waiting for I2P Service......
running: PID:8113
```

5. We will see that I2P service is up and running; now we add a proxy to our Firefox on port `4444`:



6. We can also access the I2P console at localhost `7657`:



And now a whole new world of I2P is open for us to explore.