

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И
РАДИОЭЛЕКТРОНИКИ» (ТУСУР)
Кафедра комплексной информационной безопасности электронно-вычислительных систем
(КИБЭВС)

УТВЕРЖДАЮ

заведующий каф. КИБЭВС

_____ А.А. Шелупанов

«_____» _____ 2015г.

ПАССИВНЫЙ ПЕРЕХВАТ ПАКЕТОВ ПО БЕСПРОВОДНОЙ СЕТИ ИНТЕРНЕТ (WI-FI)

Курсовая работа по дисциплине «Безопасность сетей ЭВМ»

Пояснительная записка

Выполнила:

студентка гр. 722

_____ М.В. Мейта

«_____» _____ 2015г.

Научный руководитель:

аспирант каф. КИБЭВС

_____ А.К. Новохрестов

«_____» _____ 2015г.

РЕФЕРАТ

Курсовая работа содержит 20 страниц, 22 рисунка, 6 источников.

RASPBERRY PI, UBUNTU, WI-FI, TCPDUMP, PCAP, RASPBIAN, WIRESHARK, BASH, СНИФФЕР, СЕТЕВОЙ ТРАФИК.

Цель работы — разработать и настроить систему пассивного перехвата пакетов по беспроводной сети Интернет (Wi-Fi) в научно-образовательных целях.

Проект выполнен с использованием следующих программных и аппаратных средств:

- ОС Linux Ubuntu 15.04;
- ОС Linux Raspbian Jessie Light 4.1;
- Wi-Fi-адаптер TP-Link TL-WN722NC;
- tcpdump — консольная утилита Unix для перехвата и анализа сетевого трафика;
- Wireshark — программа-анализатор сетевого трафика с графическим пользовательским интерфейсом.

Пояснительная записка выполнена согласно образовательному стандарту ВУЗа ОС ТУСУР 01-2013 [1] при помощи системы компьютерной вёрстки L^AT_EX.

Содержание

Введение	6
1 Используемые программные и аппаратные средства	7
2 Проектирование и настройка пакетного сниффера	8
2.1 Установка ОС Raspbian	8
2.2 Настройка проводного соединения (Ethernet)	9
2.3 Установка ssh-соединения	11
2.4 Установка драйвера TP-Link	12
2.5 Настройка режима перехвата пакетов (monitor mode)	14
2.6 Перехват трафика	17
Заключение	19
Список использованных источников	20

Введение

В настоящее время все больше устройств имеют возможность беспроводного выхода в сеть Интернет. Wi-Fi-соединение используется повсеместно за счет его удобства и развития беспроводных технологий. В связи с нарастающим применением беспроводных устройств внутри современных организаций необходимо построение системы безопасности не только корпоративных локальных сетей стандарта Ethernet, но также и беспроводных сетей стандарта Wi-Fi.

Пакетные снифферы (устройства перехвата сетевых пакетов) могут быть использованы для анализа трафика в режиме реального времени либо же сравнения ожидаемого сетевого трафика и перехваченного. Данные устройства также могут быть использованы для сбора статистики, обнаружения сетевых атак или посторонних беспроводных устройств.

Пакетный сниффер на основе миникомпьютера Raspberry Pi и свободно распространяемого программного обеспечения может стать недорогой альтернативой подобного рода устройств.

В качестве задания на курсовую работу была поставлена задача разработать пассивный перехватчик пакетов на базе Raspberry Pi, Wi-Fi-адаптера и существующих программ для перехвата и анализа сетевого трафика.

1 Используемые программные и аппаратные средства

Устройства, необходимые для выполнения данной работы:

- SD-карта памяти;
- Raspberry Pi — одноплатный миникомпьютер;
- Ethernet-кабель;
- Wi-Fi-адаптер TP-LINK TL-WN722N/NC;
- micro-USB питание для Raspberry Pi;
- основной компьютер (ноутбук) с установленной ОС Ubuntu.

Система пассивного перехвата сетевого трафика будет представлять собой миникомпьютер Raspberry Pi с Wi-Fi-адаптером, который будет соединен с основной машиной по Ethernet-кабелю (рис. 1.1). Наличие проводного соединения является необходимым только для настройки перехватчика пакетов, после чего сниффер может работать автономно в нужной Wi-Fi-точке, а также для копирования и анализа полученного дампа-файла.

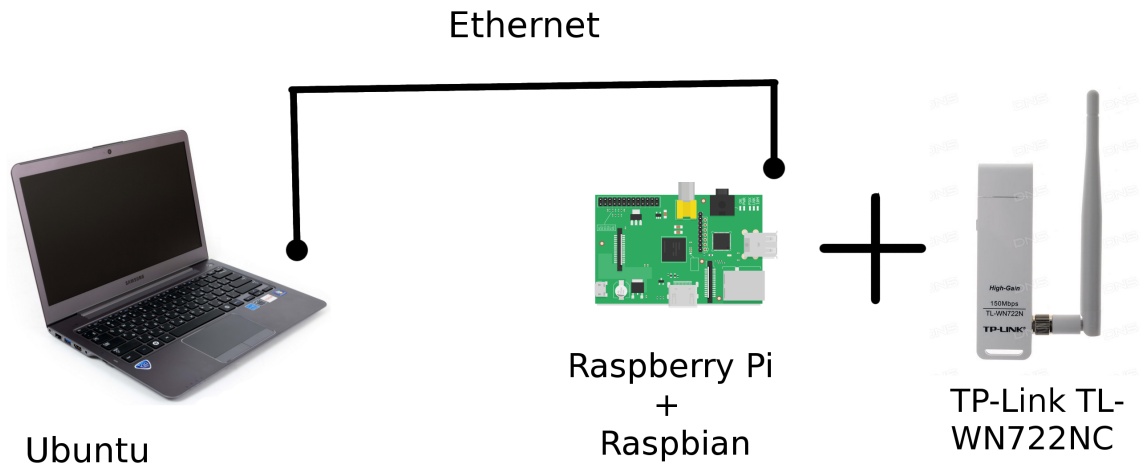


Рисунок 1.1 – Схема системы перехвата трафика

2 Проектирование и настройка пакетного sniffера

2.1 Установка ОС Raspbian

Для установки операционной системы Raspbian на Raspberry Pi необходимо перейти на страницу загрузок на официальном сайте [2] и скачать необходимый образ ОС (рис. 2.1), затем установить загруженный образ на SD-карту, которая впоследствии будет подключена к Raspberry Pi и с которой непосредственно будет загружаться система.



Рисунок 2.1 – Официальная страница загрузок ОС Raspbian

Инструкцию по установке образа системы на SD-карту можно найти, перейдя по ссылке [3].

После распаковки загруженного архива с образом системы, необходимо вставить SD-карту в слот и выполнить следующие команды:

```
$ df -h    # увидеть все примонтированные устройства
$ umount /dev/<ИМЯ_УСТРОЙСТВА>    # отмонтировать SD-карту
$ dd bs=4M if=2015-11-21-raspbian-jessie.img of=/dev/sdd    # записать образ
$ sync
```

Далее достаточно извлечь SD-карту и установить ее в соответствующий разъем Raspberry Pi.

2.2 Настройка проводного соединения (Ethernet)

Для того, чтобы удаленно зайти в систему на Raspberry Pi, необходимо подключить к нему питание (система автоматически загрузится с SD-карты), подсоединиться по Ethernet-кабелю и осуществить необходимые настройки во вкладке «Параметры системы» — «Сеть» — «Проводное» — «Параметры». Установить параметры IPv4 и IPv6, как это продемонстрировано на рисунках 2.2 и 2.3 соответственно. Все остальные настройки оставить по умолчанию.

Результатом станет рабочее проводное соединение (рис. 2.4).

Изменение Проводное соединение 1

Название соединения: Проводное соединение 1

Общий Ethernet Защита 802.1x DCB Параметры IPv4 Параметры IPv6

Способ настройки: Предоставить сеть другим компьютерам

Адреса

Адрес	Маска сети	Шлюз

Добавить
Удалить

Серверы DNS:

Поисковый домен:

ID клиента DHCP:

☒ Требовать адресацию IPv4 для этого соединения

Маршруты...

Отменить Сохранить

Рисунок 2.2 – Параметры IPv4 для проводного соединения

Изменение Проводное соединение 1

Название соединения: Проводное соединение 1

Общий Ethernet Защита 802.1x DCB Параметры IPv4 Параметры IPv6

Способ настройки: Игнорировать

Адреса

Адрес	Префикс	Шлюз

Добавить
Удалить

Серверы DNS:

Поисковый домен:

IPv6 privacy extensions: Выключено

☐ Требовать адресацию IPv6 для этого соединения

Маршруты...

Отменить Сохранить

Рисунок 2.3 – Параметры IPv6 для проводного соединения

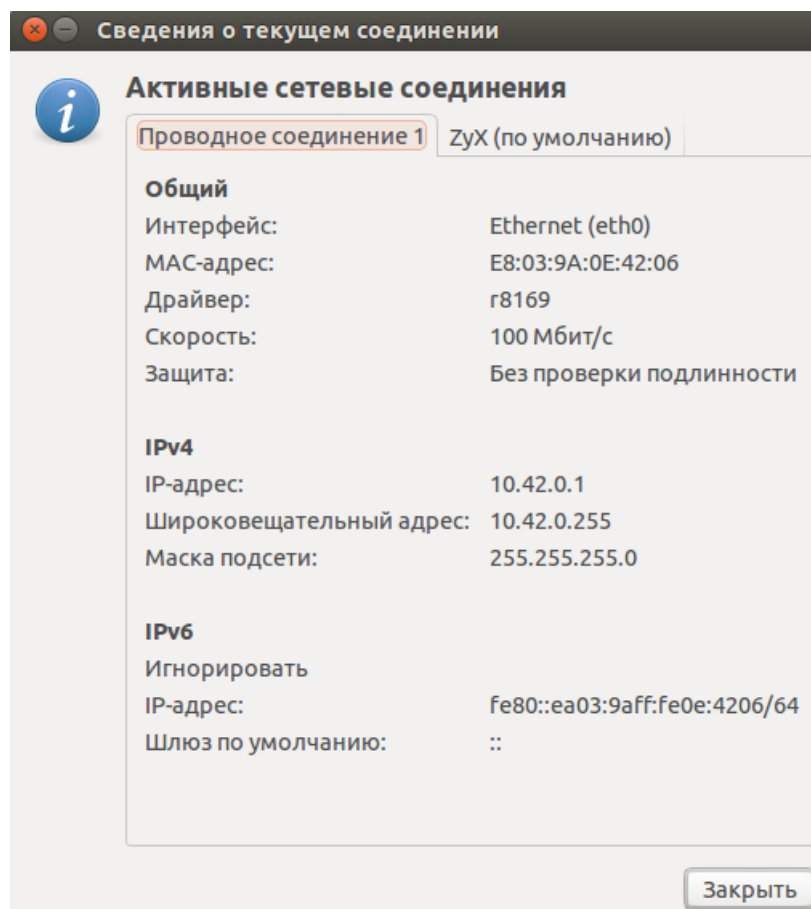


Рисунок 2.4 – Сведения о проводном соединении

2.3 Установка ssh-соединения

Для начала выведем список доступных сетевых интерфейсов из ARP-кэша при помощи следующей команды:

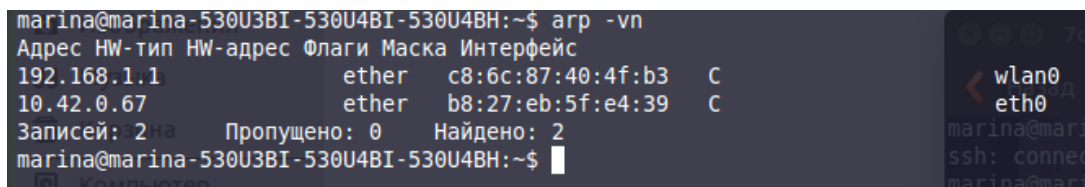
```
$ arp -vn
```

Результат выполнения команды представлен на рисунке 2.5.

Из полученной информации можно сделать вывод, что Raspberry Pi имеет IP-адрес в локальной сети 10.42.0.67. Для того, чтобы удаленно подсоединиться к нему по ssh, необходимо выполнить команду:

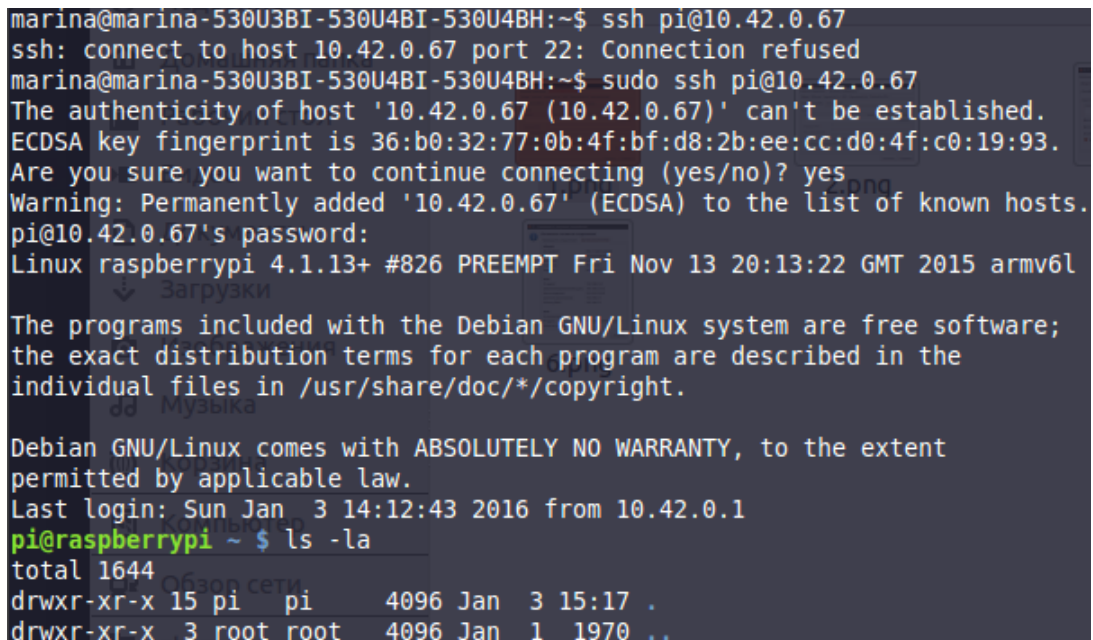
```
$ ssh pi@10.42.0.67
```

Далее система запрашивает пароль от Raspberry, после чего устанавливается ssh-соединение (рис. 2.6).



```
marina@marina-530U3BI-530U4BI-530U4BH:~$ arp -vn
Адрес HW-тип HW-адрес Флаги Маска Интерфейс
192.168.1.1 ether c8:6c:87:40:4f:b3 C
10.42.0.67 ether b8:27:eb:5f:e4:39 C
Записей: 2 На Пропущено: 0 Найдено: 2
marina@marina-530U3BI-530U4BI-530U4BH:~$
```

Рисунок 2.5 – Результат выполнения команды `arp -vn`



```
marina@marina-530U3BI-530U4BI-530U4BH:~$ ssh pi@10.42.0.67
ssh: connect to host 10.42.0.67 port 22: Connection refused
marina@marina-530U3BI-530U4BI-530U4BH:~$ sudo ssh pi@10.42.0.67
The authenticity of host '10.42.0.67 (10.42.0.67)' can't be established.
ECDSA key fingerprint is 36:b0:32:77:0b:4f:bf:d8:2b:ee:cc:d0:4f:c0:19:93.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.42.0.67' (ECDSA) to the list of known hosts.
pi@10.42.0.67's password:
Linux raspberrypi 4.1.13+ #826 PREEMPT Fri Nov 13 20:13:22 GMT 2015 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Jan 3 14:12:43 2016 from 10.42.0.1
pi@raspberrypi ~ $ ls -la
total 1644
drwxr-xr-x 15 pi pi 4096 Jan 3 15:17 .
drwxr-xr-x 3 root root 4096 Jan 1 1970 ..
```

Рисунок 2.6 – Установка ssh-соединения

2.4 Установка драйвера TP-Link

Необходимо узнать (проверить) версию системы (Raspbian) с помощью команды (рис. 2.7):

```
$ uname -a
```

Далее проверить список доступных USB-устройств (рис. 2.7). Запись «ID 0BDA:8179» и версия ОС будут определять, какой драйвер необходим.

```
pi@raspberrypi / $ uname -a
Linux raspberrypi 4.1.13+ #826 PREEMPT Fri Nov 13 20:13:22 GMT 2015 armv6l GNU/Linux
```

Рисунок 2.7 – Текущая версия системы

```
pi@raspberrypi / $ lsusb
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
Bus 001 Device 004: ID 0bda:8179 Realtek Semiconductor Corp.
```

Рисунок 2.8 – Список USB-устройств

Подробную инструкцию можно найти, перейдя по ссылке [4].

Остается загрузить необходимый драйвер (рис. 2.9), установить его и перезагрузиться (рис. 2.10). В случае успешной установки драйвера, команда `ifconfig` отобразит наличие беспроводного соединения `wlan0` (рис. 2.11).

```
pi@raspberrypi ~ $ wget https://dl.dropboxusercontent.com/u/80256631/8188eu-20151113.tar.gz
--2016-01-03 16:26:02-- https://dl.dropboxusercontent.com/u/80256631/8188eu-20151113.tar.gz
Resolving dl.dropboxusercontent.com (dl.dropboxusercontent.com)... 199.47.217.69
Connecting to dl.dropboxusercontent.com (dl.dropboxusercontent.com)|199.47.217.69|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 416724 (407K) [application/octet-stream]
Saving to: '8188eu-20151113.tar.gz'

100%[=====] 416.72K 221 KB/s in 1.9s

2016-01-03 16:26:11 (221 KB/s) - '8188eu-20151113.tar.gz' saved [416724/416724]
```

Рисунок 2.9 – Загрузка драйвера

```
pi@raspberrypi ~ $ tar -zxvf 8188eu-20151113.tar.gz
8188eu.ko
8188eu.conf
install.sh
pi@raspberrypi ~ $ ./install.sh
sudo cp 8188eu.conf /etc/modprobe.d/.
sudo install -p -m 644 8188eu.ko /lib/modules/4.1.13+/kernel/drivers/net/wireless
sudo depmod 4.1.13+

Reboot to run the driver.

If you have already configured your wifi it should start up and connect to your wireless network.

If you have not configured your wifi you will need to do that to enable the wifi.
pi@raspberrypi ~ $ sudo reboot
```

Рисунок 2.10 – Установка драйвера

```

pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:5f:e4:39
          inet addr:10.42.0.67  Bcast:10.42.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:169 errors:0 dropped:0 overruns:0 frame:0
          TX packets:155 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:19389 (18.9 KiB)  TX bytes:21375 (20.8 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

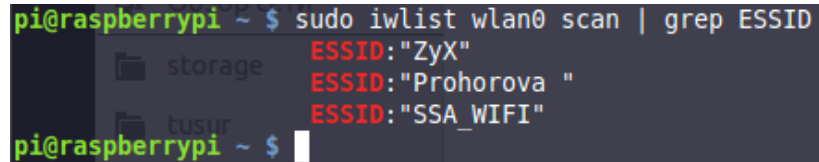
wlan0     Link encap:Ethernet  HWaddr c0:4a:00:25:6e:40
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

```

Рисунок 2.11 – Результат выполнения команды ifconfig

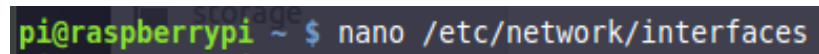
2.5 Настройка режима перехвата пакетов (monitor mode)

Для начала проверим работоспособность адаптера, настроив его на нужное соединение. Это можно осуществить, просканировав сеть (рис. 2.12) на доступные беспроводные соединения, выбрав необходимое и прописав настройки в файле `/etc/network/interfaces` (рис. 2.13 и 2.14). В результате выполнения команды `iwconfig` (рис. 2.15) можно увидеть, что уровень сигнала `wlan0` нулевой, однако после перезагрузки системы командой `reboot` сеть работает в режиме по умолчанию — `managed` (рис. 2.16).



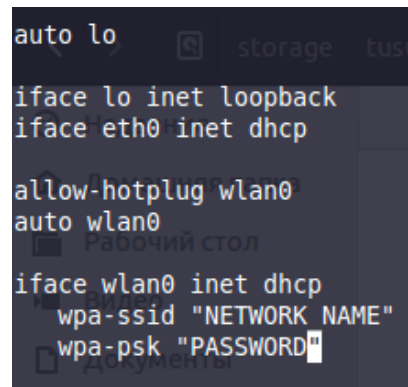
```
pi@raspberrypi ~ $ sudo iwlist wlan0 scan | grep ESSID
ESSID:"ZyX"
ESSID:"Prohorova "
ESSID:"SSA_WIFI"
pi@raspberrypi ~ $
```

Рисунок 2.12 – Сканирование сети на доступные соединения



```
pi@raspberrypi ~ $ nano /etc/network/interfaces
```

Рисунок 2.13 – Путь к config-фалу с настройками интерфейсов

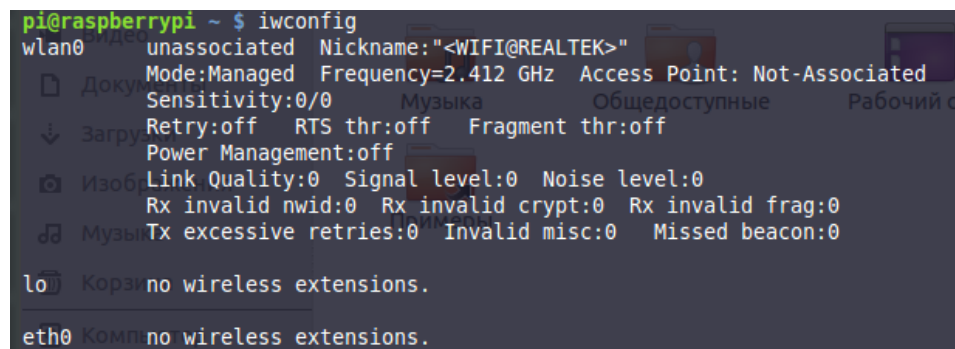


```
auto lo
iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
auto wlan0

iface wlan0 inet dhcp
    wpa-ssid "NETWORK NAME"
    wpa-psk "PASSWORD"
```

Рисунок 2.14 – Файл «interfaces»



```
pi@raspberrypi ~ $ iwconfig
wlan0 unassociated Nickname:"<WIFI@REALTEK>"
      Mode:Managed Frequency=2.412 GHz Access Point: Not-Associated
      Sensitivity:0/0
      Retry:off RTS thr:off Fragment thr:off
      Power Management:off
      Link Quality:0 Signal level:0 Noise level:0
      Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
      Tx excessive retries:0 Invalid misc:0 Missed beacon:0

lo Корзина wireless extensions.
eth0 Корзина wireless extensions.
```

Рисунок 2.15 – Результат выполнения команды `iwconfig`

```

pi@raspberrypi ~$ iwconfig
wlan0 IEEE 802.11bgn ESSID:"ZyX" Nickname:"<WIFI@REALTEK>"
Mode:Managed Frequency:2.412 GHz Access Point: C8:6C:87:40:4F:B3
Bit Rate:150 Mb/s Sensitivity:0/0
Retry:off RTS thr:off Fragment thr:off
Power Management:off
Link Quality=100/100 Signal level=59/100 Noise level=0/100
Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
Tx excessive retries:0 Invalid misc:0 Missed beacon:0
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
lo no wireless extensions.
lo Link encap:Local Loopback
eth0 no wireless extensions.

```

Рисунок 2.16 – Результат выполнения команды `iwconfig` после перезагрузки системы

Беспроводные сетевые адаптеры стандарта WNIC 802.11 могут поддерживать шесть рабочих режимов: master, managed, ad-hoc, mesh, repeater и monitor. Режим мониторинга пакетов (monitor mode) аналогичен смешанному режиму (promiscuous mode), однако применим только для беспроводных сетей. В отличие от смешанного режима, устройства не обязательно должны быть в сети. Режим мониторинга пакетов позволяет перехватывать все пакеты, которые могут быть распознаны WNIC-адаптером. Данный режим зависит от драйвера, архитектуры и чипсета устройства. Согласно этим ограничениям, не все адаптеры поддерживают monitor mode. При этом устройство может в определенный момент времени находиться только в одном режиме [5].

Настроим адаптер в режим мониторинга пакетов. Для этого необходимо ввести команду:

```
$ sudo iwconfig wlan0 mode monitor
```

При попытке ввести данную команду система выводит сообщение о том, что ресурс занят (рис. 2.17).

```

pi@raspberrypi ~$ sudo iwconfig wlan0 mode monitor
Error for wireless request "Set Mode" (8B06) :
SET failed on device wlan0 ; Device or resource busy.

```

Рисунок 2.17 – Результат выполнения команды `iwconfig`

Это связано с тем, что в системе работает network interface plugging daemon [6] — программа, работающая в системе в фоновом режиме и отвечающая за автоматическое подключение сетевых интерфейсов. Эта же программа устанавливает для устройств режим по умолчанию. Необходимо ее отключить для того, чтобы перевести устройство в другой режим работы.

Напишем небольшой bash-скрипт, который будет переводить адаптер в режим перехвата пакетов и содержать следующие команды:

```

$ #!/bin/bash
$ sudo service ifplugd stop #останавливаем работу демона
$ sudo ifconfig wlan0 down #отключаем wi-fi соединение
$ sudo iwconfig wlan0 mode monitor #включаем прослушивающий режим
$ sudo ifconfig wlan0 up #включаем wi-fi соединение
$ sudo service ifplugd start #запускаем демона
$ iwconfig #проверяем настройки

```

Результат работы скрипта приведен на рисунке 2.18. Wi-Fi-адаптер теперь работает в режиме перехвата пакетов. Данный скрипт необходимо запускать снова при перезапуске системы.

```

pi@raspberrypi ~/sniff $ cat start_monitoring.sh
#!/bin/bash

sudo service ifplugd stop
sudo ifconfig wlan0 down
sudo iwconfig wlan0 mode monitor
sudo ifconfig wlan0 up
iwconfig

pi@raspberrypi ~/sniff $ ./start_monitoring.sh
[ ok ] Network Interface Plugging Daemon...stop eth0...stop wlan0...done.
wlan0 IEEE 802.11bgn Mode:Monitor Frequency:2.412 GHz Tx-Power=20 dBm
Retry short limit:7 RTS thr:off Fragment thr:off
Power Management:off

lo no wireless extensions.

eth0 no wireless extensions.

pi@raspberrypi ~/sniff $

```

Рисунок 2.18 – Результат выполнения скрипта

2.6 Перехват трафика

Tcpdump — это консольная утилита Unix для перехвата и анализа сетевого трафика. Необходимо при запуске программы указать ей сетевой интерфейс, который будет прослушиваться. В нашем случае это Wi-Fi-адаптер (wlan0).

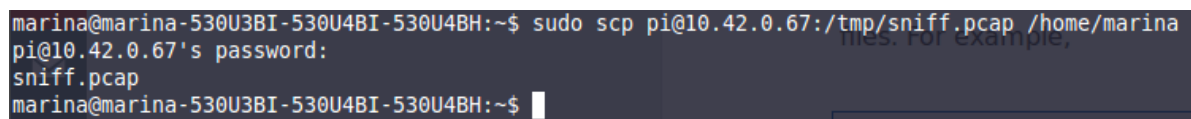
Напишем bash-скрипт со следующей командой:

```
$ #!/bin/bash
$ sudo tcpdump -v -i wlan0 -s 0 -w /tmp/sniff.pcap -c <число_пакетов>
```

Параметр -с задает максимальное количество пакетов для перехвата, после чего система останавливает выполнение скрипта.

Результат выполнения tcpdump записывается в файл /tmp/sniff.pcap. Pcap — это универсальный формат, который позволит в дальнейшем проанализировать перехваченный трафик в других программах.

После этого необходимо скопировать полученный pcap-файл на основную машину с Ubuntu для дальнейшего анализа в программе Wireshark. Сделать это можно при помощи команды scp (рис. 2.19).



```
marina@marina-530U3BI-530U4BI-530U4BH:~$ sudo scp pi@10.42.0.67:/tmp/sniff.pcap /home/marina
pi@10.42.0.67's password:
sniff.pcap
marina@marina-530U3BI-530U4BI-530U4BH:~$
```

Рисунок 2.19 – Копирование pcap-файла по ssh-соединению

Теперь на главной машине можно открыть полученный pcap-файл в программе Wireshark и ознакомиться с его содержимым (рис. 2.20). Tcpdump можно также использовать для чтения и анализа полученного pcap-файла (рис. 2.21).

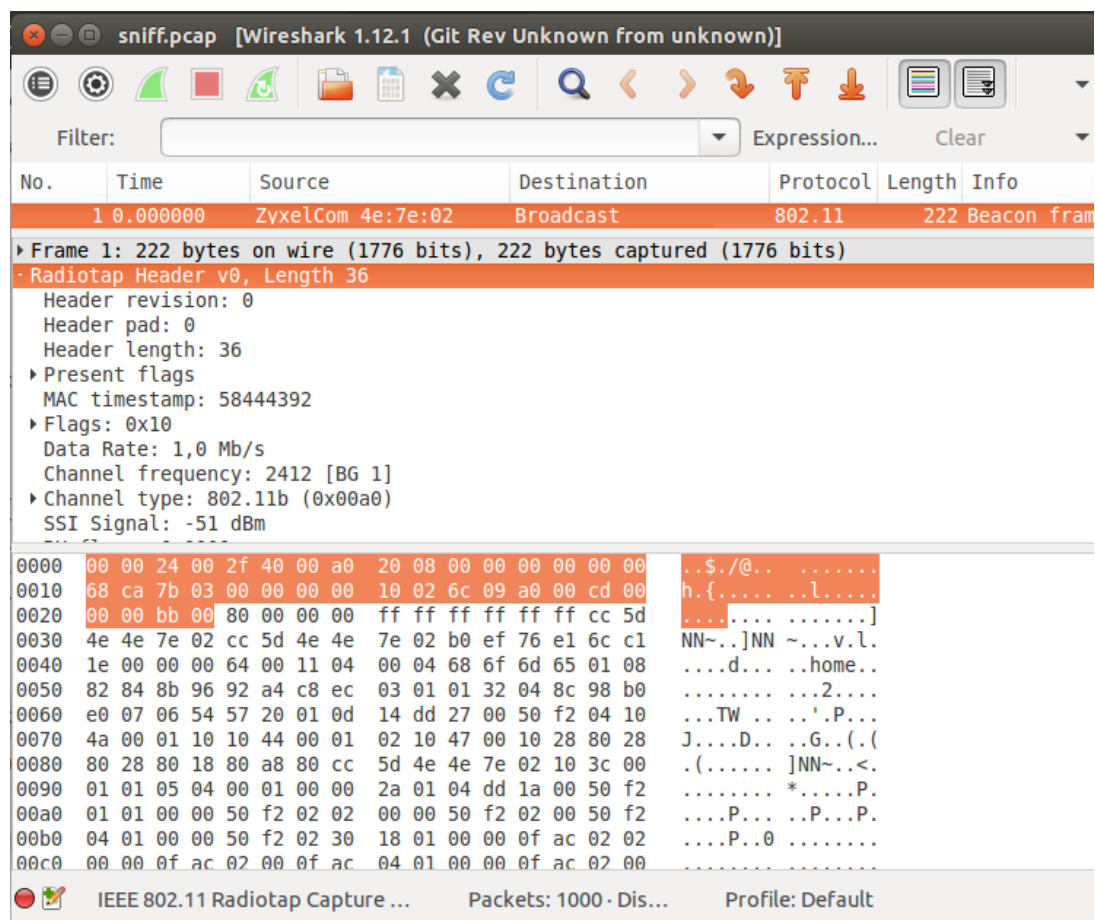


Рисунок 2.20 – Просмотр содержимого pcap-файла в wireshark

```

marina@marina-530U3B1-530U4B1-530U4B1:/storage/tusur/7 semester/ECB3M кыпсак/raspberry packet sniffer$ tcpdump -tttnr sniff_plain.pcap
reading from file sniff_plain.pcap, link-type IEEE802.11 RADIO (802.11 plus radiotap header)
2016-01-06 21:41:54.443511 36131553905us tsft 1.0 Mb/s 2412 MHz 11b -92dB signal [bit 29] Beacon (tatyana) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.454135 36131564408us tsft 1.0 Mb/s 2412 MHz 11b -92dB signal [bit 29] Beacon (otlnal) [1.0* 2.0* 5.5* 11.0* 6.0 9.0 12.0 18.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.475218 36131815427us tsft 1.0 Mb/s 2412 MHz 11b -71dB signal [bit 29] Beacon (Bukhara Rostelecom) [1.0* 2.0* 5.5* 11.0* 18.0 24.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.479130 36131818986us tsft 1.0 Mb/s 2412 MHz 11b -94dB signal [bit 29] Beacon (Wikita) [1.0* 2.0* 5.5* 11.0* 9.0 18.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.524600 36131668127us tsft 1.0 Mb/s 2412 MHz 11b -81dB signal [bit 29] Beacon (Zyx) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.545871 36131689974us tsft 1.0 Mb/s 2412 MHz 11b -93dB signal [bit 29] Beacon (tatyana) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.556496 36131666796us tsft 1.0 Mb/s 2412 MHz 11b -92dB signal [bit 29] Beacon (otlnal) [1.0* 2.0* 5.5* 11.0* 6.0 9.0 12.0 18.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.579972 36131688983us tsft 1.0 Mb/s 2412 MHz 11b -69dB signal [bit 29] Beacon (Bukhara Rostelecom) [1.0* 2.0* 5.5* 11.0* 18.0 24.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.581634 36131694109us tsft 24.0 Mb/s 2412 MHz 11g -67dB signal [bit 29] Clear-To-Send RA:c8:6c:87:40:4f:b3
2016-01-06 21:41:54.581945 36131694211us tsft 24.0 Mb/s 2412 MHz 11g -65dB signal [bit 29] BA RA:c8:6c:87:40:4f:b3
2016-01-06 21:41:54.612773 36131821378us tsft 1.0 Mb/s 2412 MHz 11b -90dB signal [bit 29] Beacon (ASUS-Val) [1.0* 2.0* 5.5* 11.0* 9.0 18.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.626986 36131737762us tsft 1.0 Mb/s 2412 MHz 11b -80dB signal [bit 29] Beacon (Zyx) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.680032 36131823634us tsft 1.0 Mb/s 2412 MHz 11b -71dB signal [bit 29] Beacon (Bukhara Rostelecom) [1.0* 2.0* 5.5* 11.0* 18.0 24.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.715218 36131825472us tsft 1.0 Mb/s 2412 MHz 11b -91dB signal [bit 29] Beacon (ASUS-Val) [1.0* 2.0* 5.5* 11.0* 9.0 18.0 36.0 54.0 Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.729384 36133984553us tsft 1.0 Mb/s 2412 MHz 11b -81dB signal [bit 29] Beacon (Zyx) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY
2016-01-06 21:41:54.750667 36133925499us tsft 1.0 Mb/s 2412 MHz 11b -95dB signal [bit 29] Beacon (tatyana) [1.0* 2.0* 5.5* 11.0* 9.0* 18.0* 36.0* 54.0* Mbit] ESS CH: 1, PRIVACY

```

Рисунок 2.21 – Просмотр содержимого pcap-файла в tcpdump

Заключение

Результатом проделанной работы стал перехватчик сетевых пакетов на основе недорогих аппаратных средств и существующих open-source-программ, пригодный для перехвата и дальнейшего анализа трафика. Сниффер может быть использован как в научно-образовательных целях, так и для обеспечения безопасности беспроводных сетей внутри какой-либо организации.

Список использованных источников

- 1 Образовательный стандарт ВУЗа ОС ТУСУР 01-2013 [Электронный ресурс]. URL: <http://asu.tusur.ru/learning/books/b12.pdf> (дата обращения: 27.09.2015).
- 2 Raspbian [Электронный ресурс]. URL: <https://www.raspberrypi.org/downloads/raspbian/> (дата обращения: 25.09.2015).
- 3 Installing Operating System Images on Linux [Электронный ресурс]. URL: <https://www.raspberrypi.org/documentation/installation/installing-images/README.md> (дата обращения: 25.09.2015).
- 4 Drivers for TL-WN725N V2 — 3.6.11+ -> 4.1.xx+ [Электронный ресурс]. URL: <https://www.raspberrypi.org/forums/viewtopic.php?p=462982> (дата обращения: 10.10.2015).
- 5 Real Time Wireless Packet Monitoring with Raspberry Pi Sniffer [Электронный ресурс]. URL: http://san.ee.ic.ac.uk/iscis2014/proceedings/27_turk.pdf (дата обращения: 19.09.2015).
- 6 Wifi ad-hoc setup raspbian [Электронный ресурс]. URL: <https://www.raspberrypi.org/forums/viewtopic.php?t=24615p=263373> (дата обращения: 09.11.2015).