



АНАЛИЗАТОР HTTP/HTTPS ТРАФИКА

Анализатор HTTP/HTTPS пакетов Техническое описание

Санкт-Петербург 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
СПЕЦИФИКАЦИЯ РЕАЛИЗОВАННЫХ МОДУЛЕЙ.....	4
БЛОК-СХЕМА ПРОГРАММЫ.....	6
РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ.....	8

ВВЕДЕНИЕ

Программный продукт предназначен для сбора и анализа HTTP/HTTPS пакетов, с целью получения информации о размерах входящего и исходящего сетевого трафика.

Приложение реализовано средствами языка программирования C++ с использованием библиотек `libcap`, `glog`, `gtest` и представляет из себя консольное приложение для работы в операционной системе Linux.

СПЕЦИФИКАЦИЯ РЕАЛИЗОВАННЫХ МОДУЛЕЙ

Исходный код приложения содержит в себе следующие функции:

- `int main(int argc, char** argv)` – точка входа в программу. Принимает на вход аргументы командой строки. Содержит в себе реализацию определения устройства, для которого происходит анализ трафика, и запуск непосредственно анализа трафика;
- `int get_dns_name(char* dns_name, in_addr ip_addr)` – определение имени по IP-адресу. Принимает на вход два аргумента: первый – строка, в которую будет записано имя; второй – адрес, для которого нужно определить имя. Возвращаемое значение определяет корректность работы функции;
- `void packet_callback(u_char *args, const struct pcap_pkthdr *header, const u_char* packet)` – callback-функция, вызываемая функцией библиотеки «libpcap» `pcap_loop`, когда получен пакет. Реализует обработку полученного пакета. Принимает на вход три аргумента: `args` – аргументы, передаваемый пользователем; `header`, `packet` – аргументы, передаваемый функцией `pcap_loop`;
- `void print_help()` – печать на экран справки о программе;
- `void cout_stat(size_t period)` – печать на экран информации о трафике. Принимает на вход один аргумент – периодичность вывода в секундах;
- `void sig_handler(int signal)` – callback-функция, вызываемая при получении программой сигнала `SIGINT`. Корректно завершает программу, высвобождая захваченные ресурсы. Принимает на вход один параметр – полученный сигнал;
- `int set_log_settings(char* argv_0)` – устанавливает настройки логгирования. Принимает на вход один аргумент – первый аргумент командной строки. Возвращаемое значение определяет корректность работы функции;

- `int arguments_check(int argc, char** argv, size_t& num_packets, char* filter_exp, size_t& period)` – проверяет полученные аргументы командной строки. Настраивает параметры программы согласно полученным значениям. Возвращаемое значение определяет корректность работы функции.

БЛОК-СХЕМА ПРОГРАММЫ

На рисунке 1 представлена блок схема функции main.

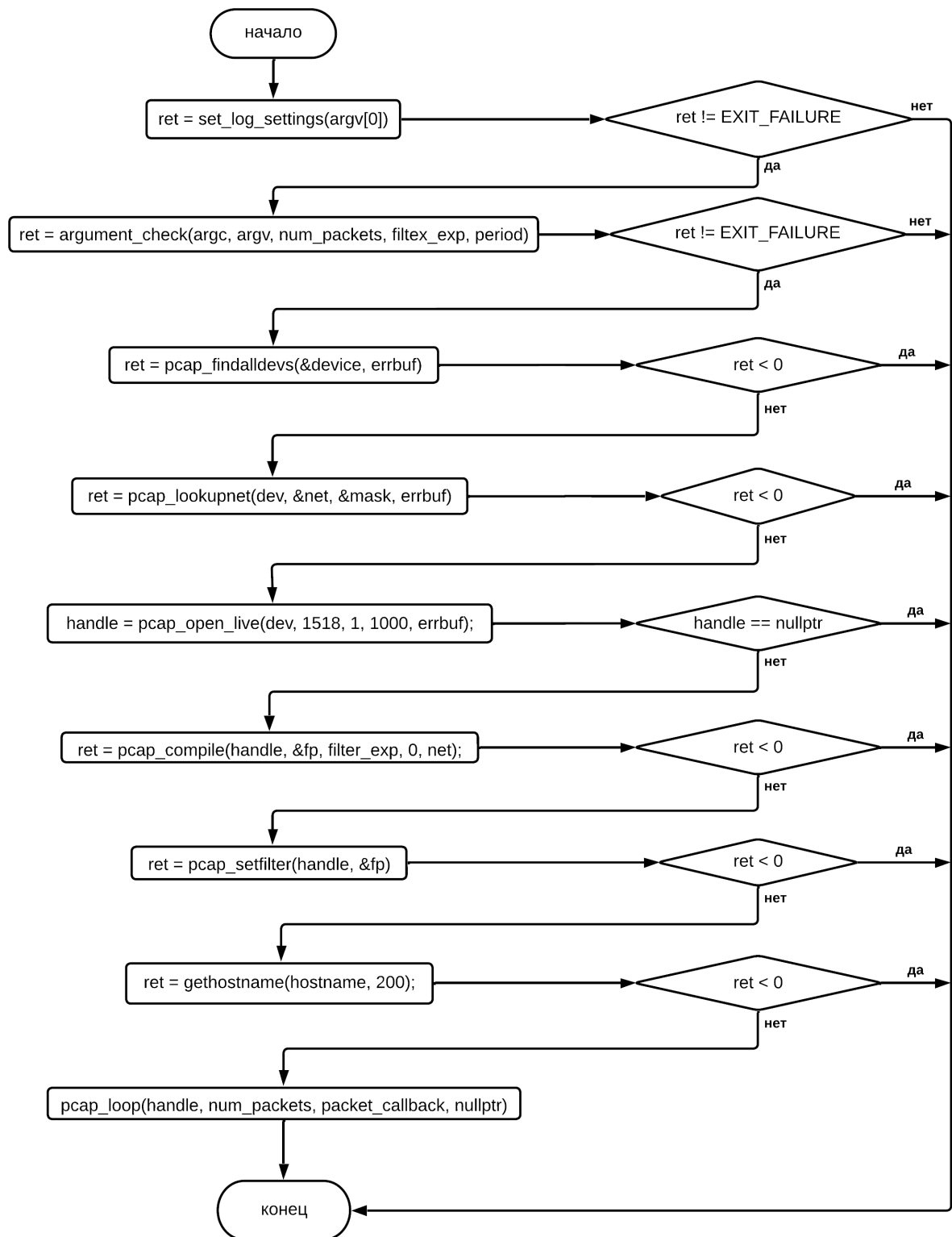


Рисунок 1 – Блок-схема функции main

На рисунке 2 представлена логическая схема функции packet_callback.

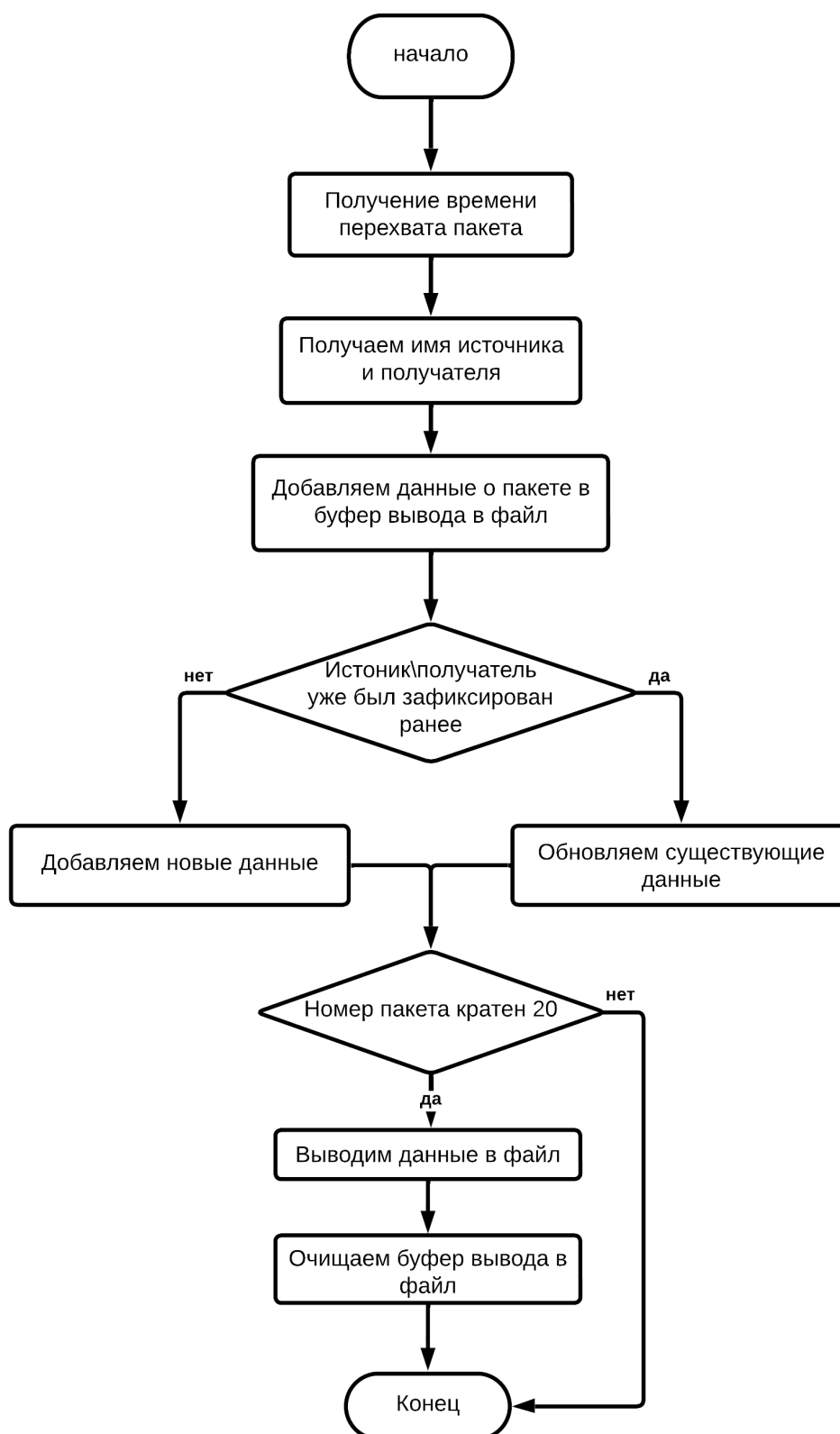


Рисунок 2 – Схема функция packet_callback

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Запуск программы осуществляется через исполняемый файл. Для этого в терминал нужно ввести путь до файла и имя файл (Для фильтра по умолчанию требуется запуск программы с правами суперпользователя). Пример запуска представлен на рисунке 3.

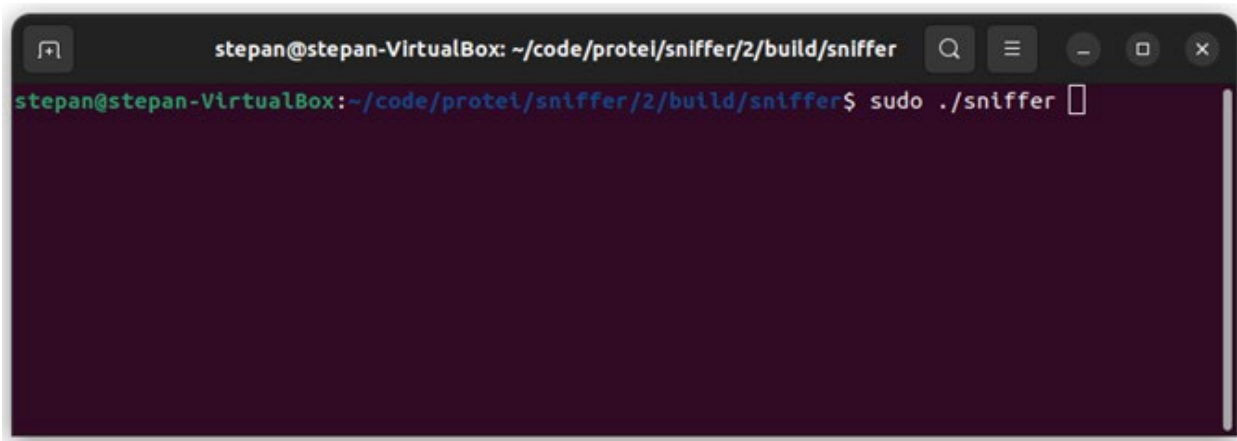
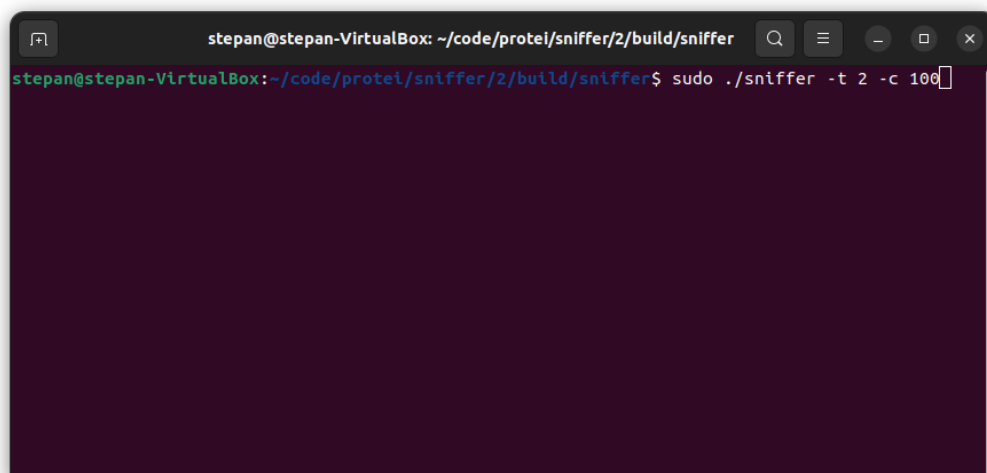


Рисунок 3 – Пример команды запуска программы

Программа поддерживает несколько параметров запуска (квадратные скобки указывают тип данных):

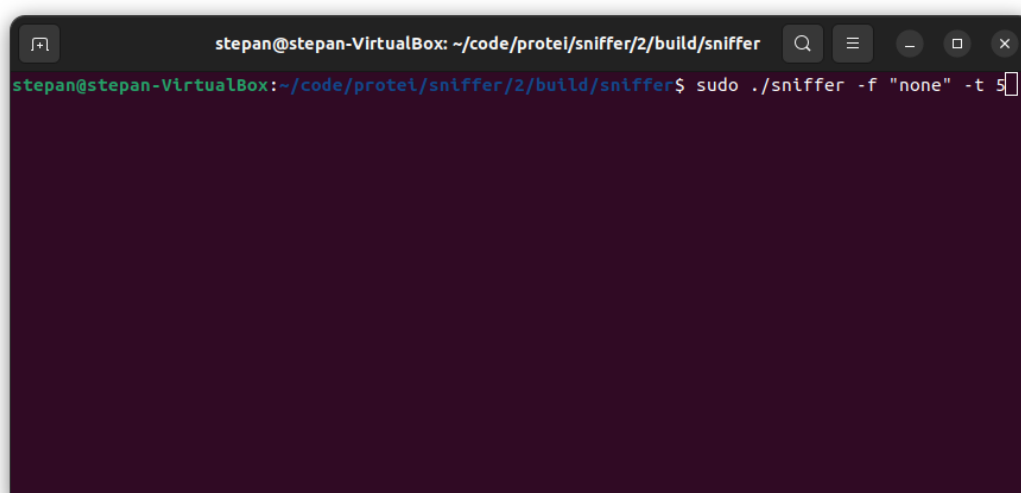
- `-t [число]` – интервал вывода данных на экран. Целое число от 0 до 9 10 включительно;
- `-c [число]` – количество пакетов для захвата. Целое неотрицательное число. По умолчанию – 0 (захват пакетов до закрытия программы);
- `-f [строка]` – задание фильтра для перехвата пакетов. Выражение указывается к двойным кавычкам;
- `-h` – вывод информации о доступных параметрах запуска.

На рисунках 4, 5 показаны примеры запуска программ с использованием параметров запуска.



```
stepan@stepan-VirtualBox: ~/code/protei/sniffer/2/build/sniffer
stepan@stepan-VirtualBox:~/code/protei/sniffer/2/build/sniffer$ sudo ./sniffer -t 2 -c 100
```

Рисунок 4 – Пример запуска программы с использованием параметров запуска



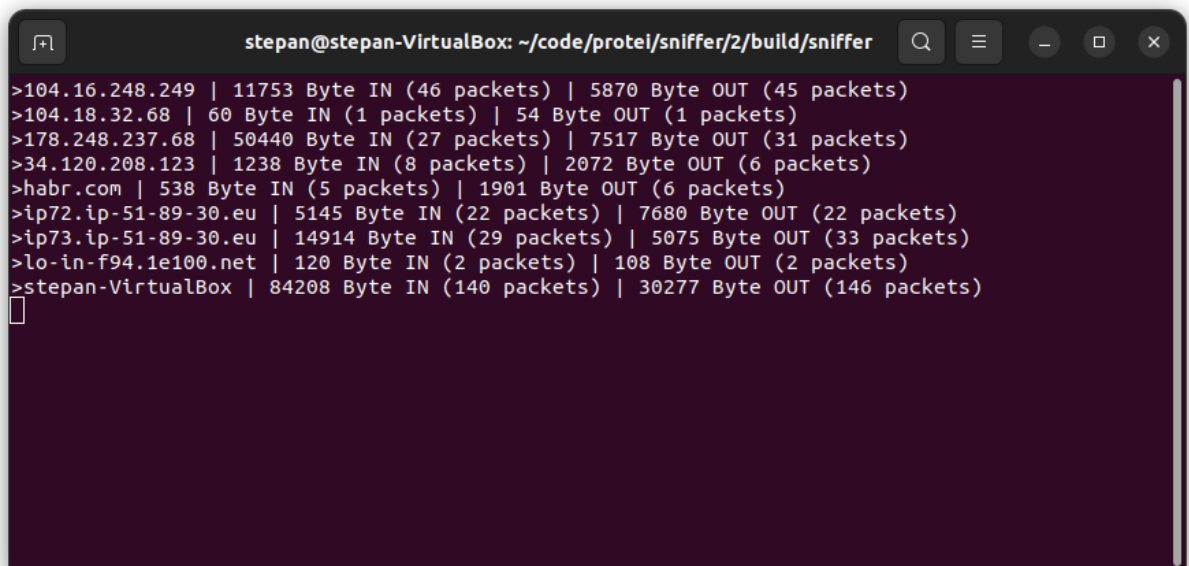
```
stepan@stepan-VirtualBox: ~/code/protei/sniffer/2/build/sniffer
stepan@stepan-VirtualBox:~/code/protei/sniffer/2/build/sniffer$ sudo ./sniffer -f "none" -t 5
```

Рисунок 5 – Пример запуска программы с использованием параметров запуска

Вывод на экран осуществляется по следующему шаблону:

[IP-адрес/имя отправителя] [входной трафик] [выходной трафик]

Пример вывода программы представлен на рисунке 6.

A screenshot of a terminal window titled 'stepan@stepan-VirtualBox: ~/code/protel/sniffer/2/build/sniffer'. The terminal displays a list of network traffic statistics for various IP addresses and domains. Each line shows the destination, the number of bytes received (IN), the number of packets received, the number of bytes sent (OUT), and the number of packets sent. The last line shows the total statistics for the 'stepan-VirtualBox' interface.

```
stepan@stepan-VirtualBox: ~/code/protel/sniffer/2/build/sniffer
>104.16.248.249 | 11753 Byte IN (46 packets) | 5870 Byte OUT (45 packets)
>104.18.32.68 | 60 Byte IN (1 packets) | 54 Byte OUT (1 packets)
>178.248.237.68 | 50440 Byte IN (27 packets) | 7517 Byte OUT (31 packets)
>34.120.208.123 | 1238 Byte IN (8 packets) | 2072 Byte OUT (6 packets)
>habr.com | 538 Byte IN (5 packets) | 1901 Byte OUT (6 packets)
>ip72.ip-51-89-30.eu | 5145 Byte IN (22 packets) | 7680 Byte OUT (22 packets)
>ip73.ip-51-89-30.eu | 14914 Byte IN (29 packets) | 5075 Byte OUT (33 packets)
>lo-in-f94.1e100.net | 120 Byte IN (2 packets) | 108 Byte OUT (2 packets)
>stepan-VirtualBox | 84208 Byte IN (140 packets) | 30277 Byte OUT (146 packets)
```

Рисунок 6 – Пример вывода на экран

Стоит обратить внимание, что на рисунке 6 в последней строке приведено общее количество входящего и исходящего трафика для данного устройства.

Выход из программы осуществляется нажатием сочетания клавиш CTRL+C.