**Министерство образования и науки Российской Федерации**

**Федерально государственное автономное образовательное**

**учреждение высшего образования**

**«Санкт-Петербургский национальный исследовательский**

**университет информационных технологий, механики и оптики»**



**Мегафакультет**: Компьютерных технологий и управления

**Факультет**: Безопасности информационных технологий

**Кафедра**: Проектирования и безопасности компьютерных систем

**Направление** (специальность): 10.03.01 «Информационная безопасность»

**Лабораторная работа №2**

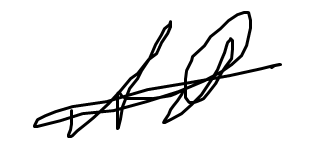
**на тему**

**«Обработка и тарификация трафика NetFlow»**

**ВАРИАНТ№ 3**

Выполнил:

Студент гр.N3352

/Распутина А.А.

Проверил:

Федоров Иван Романович

Санкт-Петербург

2020 г.

**Цель работы:**

Изучить и программно реализовать правило тарификации для услуг типа “Интернет” по размеру трафика.

**Задание:**

**Вариант № 3** **Протарифицировать абонента с IP-адресом 192.168.250.27 с коэффициентом k: 1руб/Мб.**

В данной работе предполагается обработка трафика NetFlow v5 из файла nfcapd.202002251200:

В рамках работы требуется:

1. Привести данный файл в читабельный вид (проще всего это сделать с помощью утилиты nfdump)   
   nfdump -r nfcapd.202002251200
2. Сформировать собственный файл для тарификации любого формата, с которым удобно работать (в соответствии с вариантом работы)
3. Построить график зависимости объема трафика от времени (любым удобным образом)
4. Протарифицировать трафик в соответствии с вариантом задания

**Теоретическая часть**

NetFlow — это протокол, разработанный компанией Cisco и предназначенный для сбора информации об IP-трафике внутри сети. Маршрутизаторы Cisco анализируют проходящий через интерфейс трафик, суммируют данные и отправляют статистику в формате NetFlow на специальный узел, называемый NetFlow Сollector. NetFlow часто используется для ведения биллинга или для анализа трафика сети. Протокол существует в нескольких версиях, последняя версия 9 предназначена для учёта трафика между АС (Автономная Система) и в импортируемых данных имеет несколько дополнительных полей таких как АС источника, АС назначения и пр., но обычно, для биллинга в несложной сети внутри одной АС достаточно информации, содержащейся в данных NetFlow версии ‎5.

Правила тарификации услуг “Интернет”:

X = Q \* k,

где X - итоговая стоимость, Q - общий объем трафика NetFlow за отчетный период, k - множитель тарифного плана (у каждого варианта свой).

В качестве результата работы необходимо представить программный модуль для обработки, просмотра статистики (график) и тарификации трафика NetFlow. Средства реализации выбираются студентом самостоятельно.

**Практическая часть**

Чтение файла nfcapd.202002251200 утилитой nfdump:

(если утилита отсутствует – предварительная установка - sudo apt-get install nfdump)

nfdump -r nfcapd.202002251200:

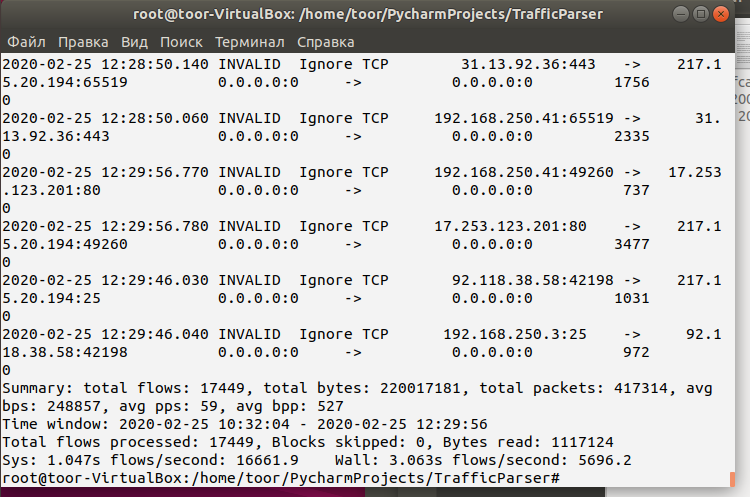


Рис. 1 – Чтение файла nfcapd.202002251200 при помощи nfdump -r nfcapd.202002251200

Перенос данных трафика в текстовый или csv файл производится командой:

nfdump -r nfcapd.202002251200 > file.txt

nfdump -r nfcapd.202002251200 > file.csv

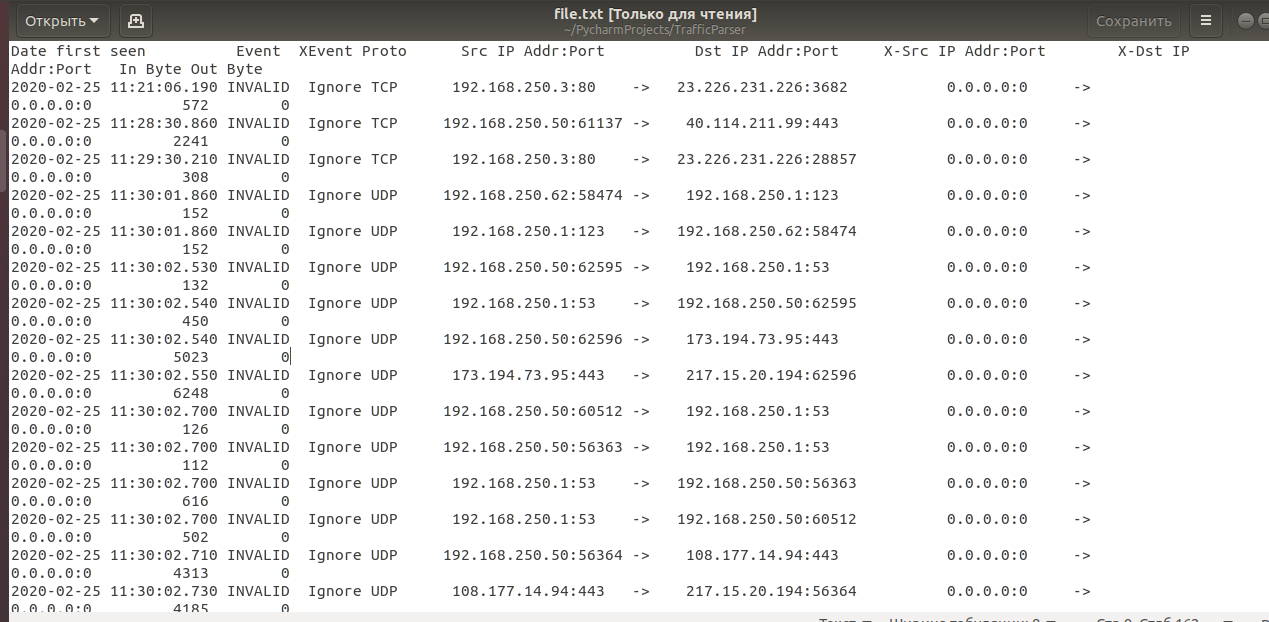


Рис. 2 – Перенос данных nfcapd.202002251200 в текстовый файл

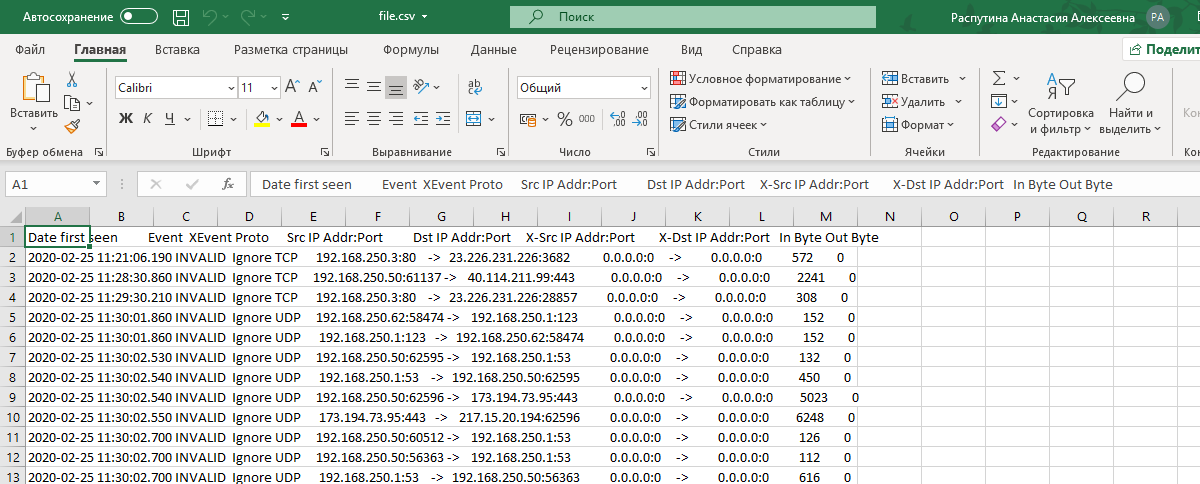


Рис. 3 - Перенос данных nfcapd.202002251200 в CSV файл

Команда для формирования файла, пригодного для парсинга:

**nfdump -r nfcapd.202002251200 -o csv > file.csv**

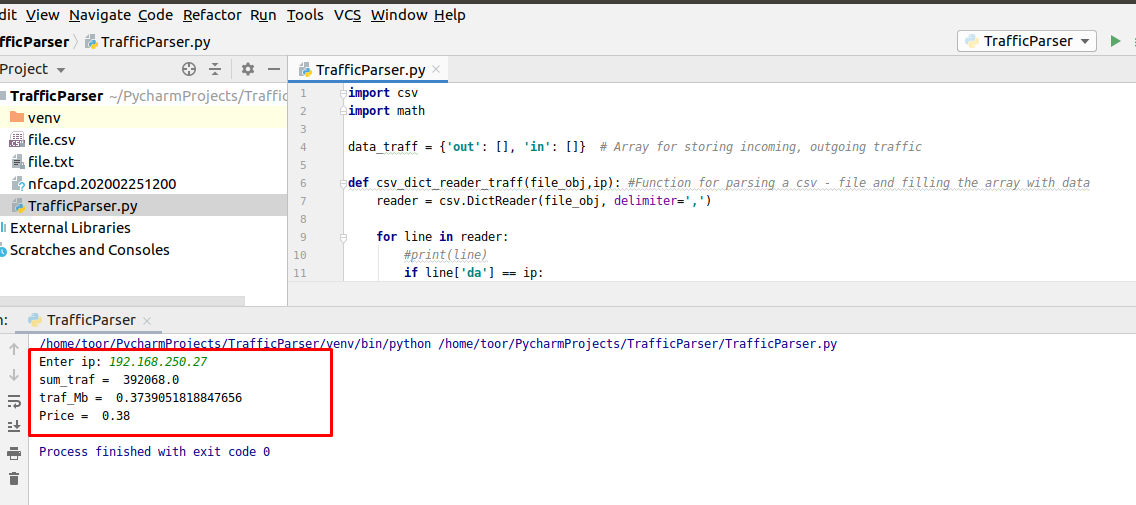
**Результат вычисления для абонента с IP-адресом 192.168.250.27 с коэффициентом k: 1руб/Мб:**

Рис. 4 – Расчет тарифа, отладка приложения в PyCharm

**Результат работы программы:**

**Enter ip: 192.168.250.27**

**sum\_traf = 392068.0 байт**

**traf\_Mb = 0.3739051818847656 Мб**

**Price = 0.38 руб**

**Проверка:**

Для проверки работы программы были проведены расчеты в Excel с фильтрами по ip из варианта и расчетом суммы к оплате за трафик по тарифу:

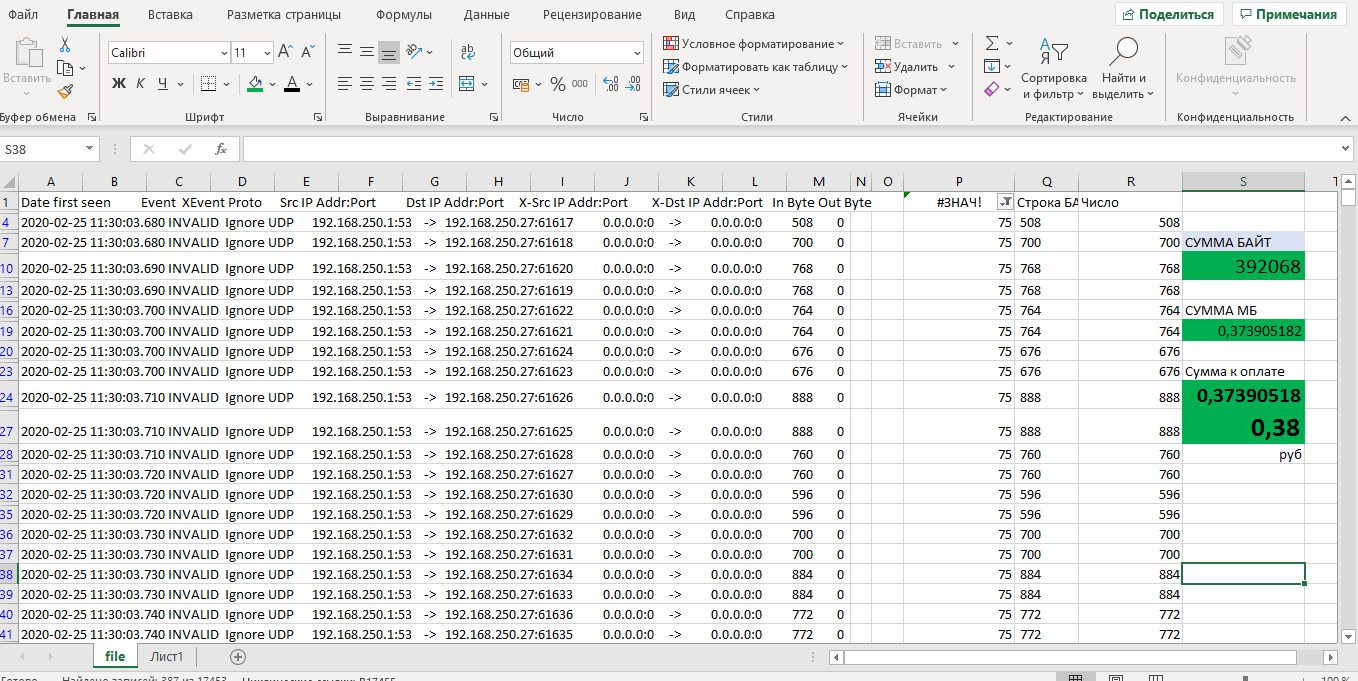


Рис. 5 – Расчет суммы по абоненту в MS Excel

**Итого к оплате: 0,38 руб, следовательно, процедура расчета отрабатывает корректно.**

**График зависимости объема трафика от времени (любым удобным образом)**

При анализе данных необходимо также учитывать, что объем данных может быть указан в разных единицах измерения:

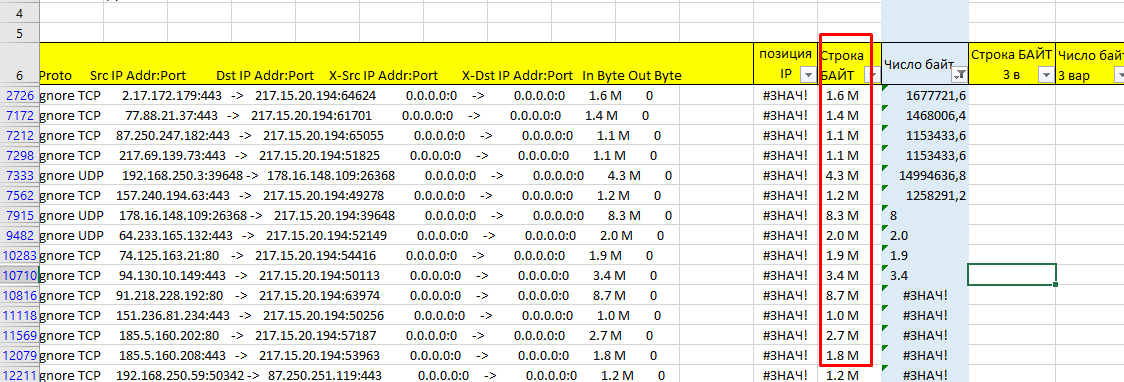
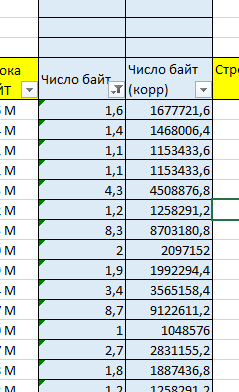
 

Рис. 6 – Обработка и анализ данных с другими единицами измерения в MS Excel

При построении графика по данным таблицы получаем результат:

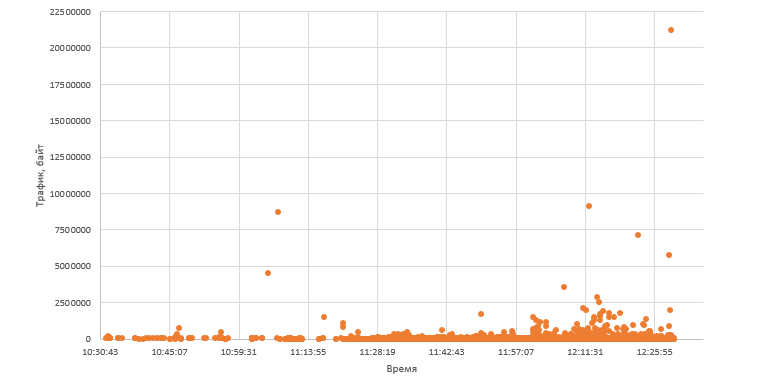


Рис. 7 – График зависимости объема трафика от времени

**Анализ трафика:**

В течение времени анализа трафика были замечены несколько значительных «пика» трафика» - в 11:06:39, 12:12:18 и в 12:29:18 – максимальное значение **- 20.2 Мб**

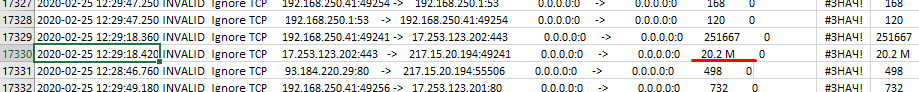


Рис. 8 – Максимальное значение объема трафика от времени

Запуск программы на исполнение производится при помощи команды из директории **TrafficParser** с файлом **TrafficParser**.py (файл с данными file.csv необходим для корректной работы программы):

**python3 TrafficParser.py** (либо python TrafficParser.py)

После запуска программы необходимо ввести ip-адрес.

В результате работы программы будет выведена сумма к оплате (если ip-адрес отсутствует в файле, будет выведена цена 0 руб):

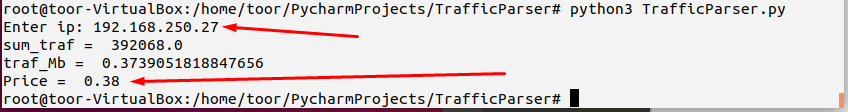


Рис. 9 – Запуск программы на исполнение и результат работы

**Выводы**

В результате проделанной работы были изучены и программно реализованы на Python правила тарификации для услуг типа “Интернет” объему трафика”.

Протарифицирован абонент с IP-адресом 192.168.250.27 с коэффициентом k: 1руб/Мб. Также построен и проанализирован график зависимости объема трафика от времени.

**Приложение**

**TrafficParser.py:**

**import** csv

**import** math

data\_traff **=** **{**'out'**:** **[],** 'in'**:** **[]}** # Array for storing incoming, outgoing traffic

**def** csv\_dict\_reader\_traff**(**file\_obj**,**ip**):** #Function for parsing a csv - file and filling the array with data

reader **=** csv**.**DictReader**(**file\_obj**,** delimiter**=**','**)**

**for** line **in** reader**:**

#print(line)

**if** line**[**'da'**]** **==** ip**:**

data\_traff**[**'in'**].**append**(**line**[**'ibyt'**])**

**if** line**[**'sa'**]** **==** ip**:**

data\_traff**[**'out'**].**append**(**line**[**'obyt'**])**

**def** traffic**(**data**):** # Payment calculation

price **=** 0

traf\_Mb **=** 0

sum\_traf **=** 0

**for** traf\_out **in** data**[**'out'**]:**

sum\_traf**+=** float**(**traf\_out**)**

**for** traf\_in **in** data**[**'in'**]:**

sum\_traf **+=** float**(**traf\_in**)**

traf\_Mb **=** sum\_traf **/** **(**2**\*\***20**)** # From bytes to Mb

**print(**'sum\_traf = ' **,** sum\_traf **)**

**print(**'traf\_Mb = ' **,** traf\_Mb **)**

price**+=**round**(**math**.**ceil**(**traf\_Mb**\***100**)/**100**,** 2**)\***1 # - 1 rub / Mb

**return** price

**with** open**(**"file.csv"**)** **as** f\_obj**:**

ip **=** input**(**"Enter ip: "**)**

csv\_dict\_reader\_traff**(**f\_obj**,** ip**)**

**print(**'Price = '**,** traffic**(**data\_traff**))**