**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

**ФАКУЛЬТЕТ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**Управление мобильными устройствами**

Лабораторная работа №2

«Обработка и тарификация трафика NetFlow»

Работу выполнил

студент 3 курса

группы N3349

Краснов Н.Д.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Федоров И.Р.

Санкт-Петербург

2020

**Цель работы:** реализовать программный модуль для обработки, просмотра статистики и тарификации трафика NetFlow.

**Описание средств реализации:** мною был выбрал язык Python, потому что у меня имеется опыт работы с ним, а также на нем необходимая задача реализуется достаточно просто.

Для удобства чтения файла, обработаем его с помощью «nfdump -r nfcapd.202002251200 -o csv > dump.csv», и вот мы имеем формат, с которым уже знаем как работать.

**Исходный код:**

Вариант 9.

Протарифицировать абонента с IP-адресом 192.168.250.3

с коэффициентом k: 0,5руб/Кб после достижения 1000Кб.

Единицы измерения выбирались в соответствии примечанию №2

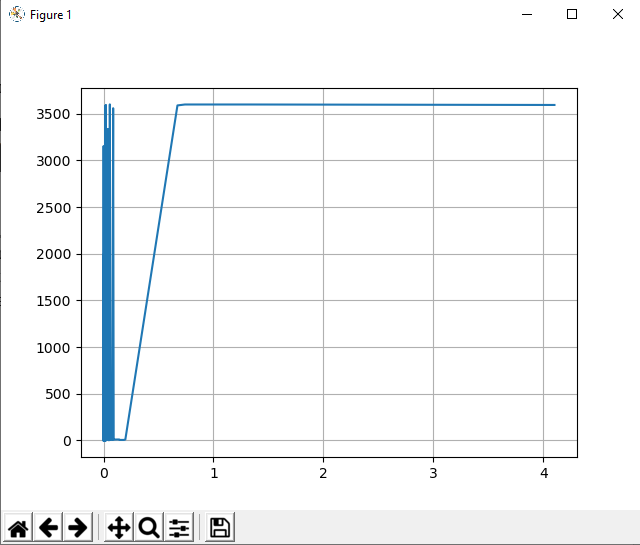
import csv  
import matplotlib.pyplot as plot  
  
cdr=[]  
bytes = 0  
price = 0  
time = []  
weight = []  
with open('nfcap.csv') as file:  
 reader = csv.reader(file)  
 for row in reader:  
 if '192.168.250.3' in row:  
 cdr.append(row)  
for i in range(len(cdr)):  
 #if (cdr[i][2])!='0.000':  
 time.append(int(float(cdr[i][2])))  
 weight.append((float(cdr[i][12]))/(2\*\*20))  
 bytes += float(cdr[i][12])  
bytes = bytes/(2\*\*10)  
if bytes > 1000:  
 price = (bytes-1000)\*0.5  
print("%.2f" % price)  
buff1 = int(0)  
buff2 = float(0)  
for i in range(len(time)-1):  
 for j in range(len(time)-i-1):  
 if weight[j] > weight[j+1]:  
 buff1 = weight[j]  
 buff2 = time[j]  
 weight[j] = weight[j+1]  
 time[j] = time[j+1]  
 weight[j+1] = buff1  
 time[j+1] = buff2  
assert len(time) == len(weight)  
plot.plot(weight,time)  
plot.grid()  
plot.show()

Результат:

По тарификации.



График, в котором ось абсцисс – объем трафика, ось ординат – длительность сеанса.



**Вывод:** при выполнении данной работы я выполнил поставленную задачу, реализовав программный модуль для обработки, просмотра статистики и тарификации трафика NetFlow.