**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«Национальный исследовательский университет ИТМО»**

**Факультет безопасности информационных технологий**

**Дисциплина:**

**«Управление мобильными устройствами»**

**Лабораторная работа № 2 на тему**

**«Обработка и тарификация трафика NetFlow»**

**Выполнил:**

Студент гр. N3349

Будянский Д.С.

**Проверил:**

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

Федоров И. Р.

Санкт-Петербург

2020г.

**Цель работы:** создать программное обеспечение для автоматизации процесса тарификации услуг «Интернет» путем обработки файла с данными статистики NetFlow версии 5.

**Теория:**

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.

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

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).

**Задание (2 вариант):** Протарифицировать абонента с IP-адресом 217.15.20.194

с коэффициентом k: 1руб/Мб, первая 1000Кб бесплатно**.** В случае, если общий объем трафика по абоненту, меньше заявленного во варианте работы (например, требуется тарификация с увеличенным коэффициентом после 1000Мб, а у абонента всего 100), то тогда уменьшается единица учета (т.е. вместо Мб считаем Кб и т д). Это наш случай

**Исходный файл**: «nfcapd.202002251200».

**Ход работы**:

Для обработки исходного файла используется команда   
«nfdump -r nfcapd.202002251200 -o csv> data.csv», сохраняющая csv вид данного файла в файл «data.csv».



Для написания ПО для тарификации был выбран язык программирования python версии 3.8. Используемые библиотеки:csv, matplotlib, datetime, itertools.

Основой выбора данного языка программирования явились: наличие библиотек для работы с файлами csv формата, построения графиков и простота понимания программного кода.

**Входные данные ПО**: «data.csv»

**Выходные данные ПО**: Количество использованных мб и их цена, график зависимости объема трафика от времени (сохраняется в виде картинки "graph.png»).

**Исходный код:**

import csv

import matplotlib

import matplotlib.pyplot as plt

ip = '217.15.20.194'

traffic\_price = 1

# 1000 Kb бесплатно т.к. требуется тарификация после 1000Мб, а у абонента меньше 1000

free\_traffic = 1

with open('data.csv', "r", newline="") as file:

reader = csv.DictReader(file, delimiter=',')

summ\_bits = 0

to\_check = []

# Обрабатываем входные данные

for row in reader:

if row['sa'] == ip or row['da'] == ip:

bits\_length = int(row['ibyt'])

summ\_bits += bits\_length

to\_check.append(('Income:' if row['sa'] == ip else 'Outcome:') + f'{bits\_length} bytes')

# Тарифицируем абонента

summ\_mb = summ\_bits / 2 \*\* 20

price = (summ\_mb - free\_traffic) \* traffic\_price if summ\_mb > free\_traffic else 0

file\_name = 'graph.png'

import datetime

import itertools

with open('data.csv', "r", newline="") as file:

reader = csv.DictReader(file, delimiter=',')

dates\_lens = []

for row in reader:

if row['sa'] == ip or row['da'] == ip:

dates\_lens.append((datetime.datetime.strptime(row['ts'], "%Y-%m-%d %H:%M:%S"), int(row['ibyt']) / 1024))

# Делаем группировку трафика по минутам

def group\_filter(x):

return datetime.datetime.strptime(x[0].strftime('%Y-%m-%d %H:%M:0'), '%Y-%m-%d %H:%M:%S')

dates\_lens = itertools.groupby(sorted(dates\_lens, key=group\_filter), group\_filter)

dates\_lens = [(key, sum([x[1] for x in group])) for key, group in dates\_lens]

dates = [x[0] for x in dates\_lens]

kb\_len = [x[1] for x in dates\_lens]

dates = matplotlib.dates.date2num(dates)

ax = plt.subplot(111)

ax.plot(dates, kb\_len)

ax.xaxis\_date()

plt.xlabel('Time')

xformatter = matplotlib.dates.DateFormatter('%H:%M')

plt.gcf().axes[0].xaxis.set\_major\_formatter(xformatter)

plt.ylabel('kB')

plt.savefig(file\_name)

# Выводим информацию

print('Mb used:', summ\_mb)

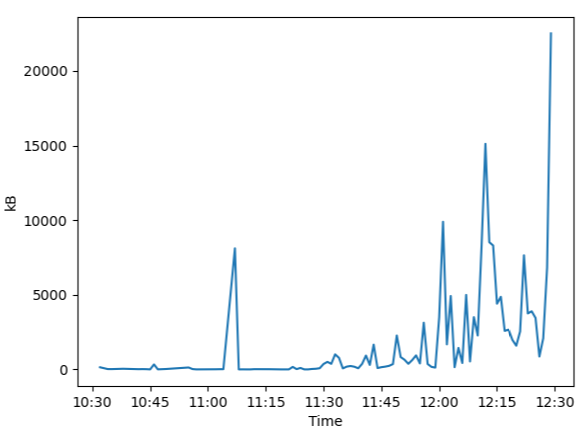
print('Price is', price, 'rubles')

print(f'Graphic image saved to current directory as: {file\_name}')

**Вывод программы**:



График (файл «graph.png»):



**Вывод**: в ходе работы была написана программа, позволяющая тарифицировать абонента и подсчитать итоговую стоимость использования услуги «Интернет» с учётом определённого тарифа, также позволяющую пострить график зависимости объема трафика от времени.