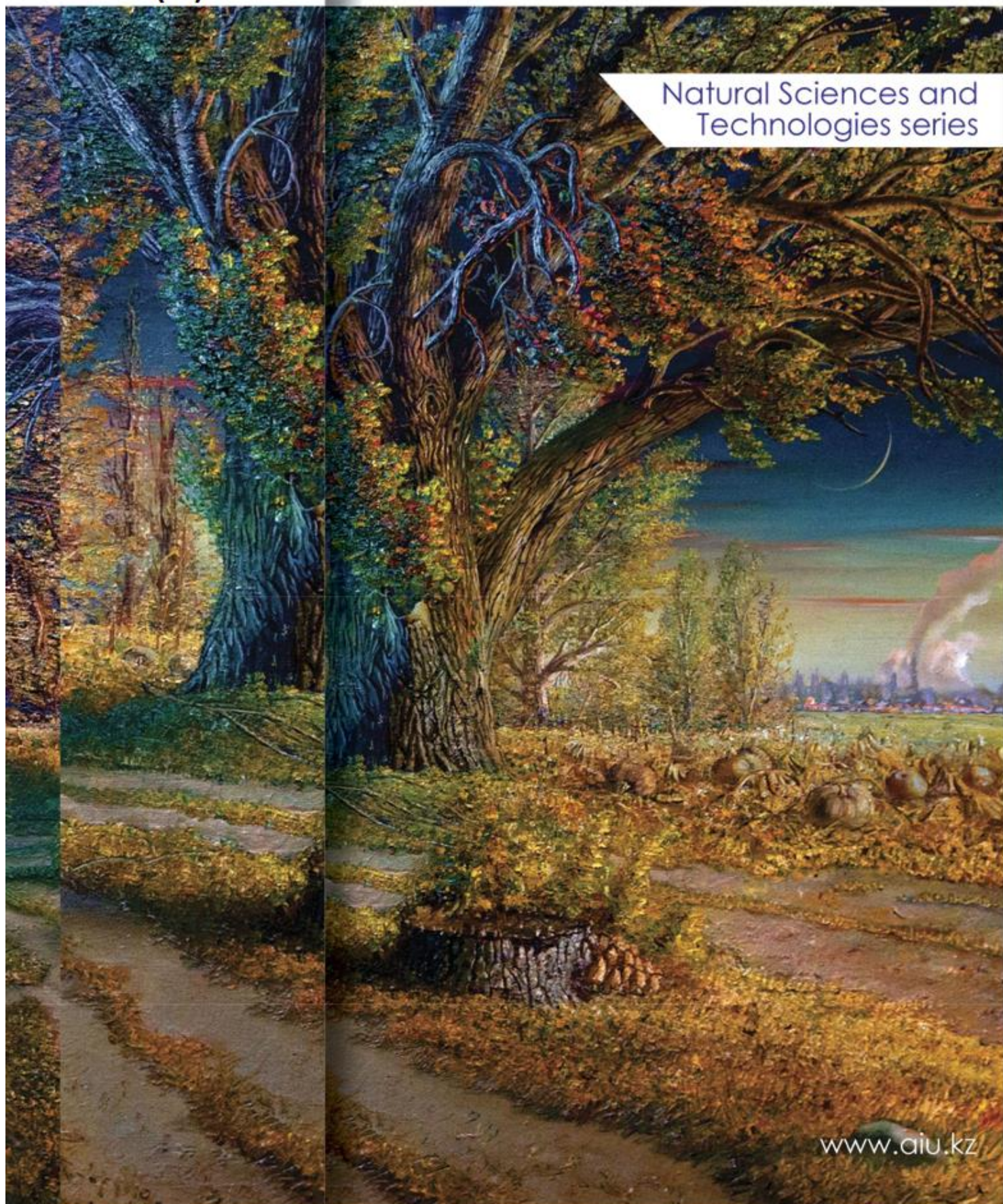


INTERNATIONAL SCIENCE REVIEWS



№1 (3) 2022

Natural Sciences and
Technologies series





INTERNATIONAL SCIENCE REVIEWS

Natural Sciences and Technologies series

Has been published since 2020

№1 (3) 2022

Nur-Sultan

EDITOR-IN-CHIEF:

Doctor of Physical and Mathematical Sciences, Academician of NAS RK, Professor
Kalimoldayev M. N.

DEPUTY EDITOR-IN-CHIEF:

Doctor of Biological Sciences, Professor
Myrzagaliyeva A. B.

EDITORIAL BOARD:

- | | |
|----------------------------|--|
| Akiyanova F. Zh. | - Doctor of Geographical Sciences, Professor (Kazakhstan) |
| Seitkan A. | - PhD, (Kazakhstan) |
| Baysholanov S. S | - Candidate of Geographical Sciences, Associate professor (Kazakhstan) |
| Zayadan B. K. | - Doctor of Biological Sciences, Professor (Kazakhstan) |
| Salnikov V. G. | - Doctor of Geographical Sciences, Professor (Kazakhstan) |
| Zhukabayeva T. K. | - PhD, (Kazakhstan) |
| Urmashiev B.A | - Candidate of Physical and Mathematical Sciences, (Kazakhstan) |
| Abdildayeva A. A. | - PhD, (Kazakhstan) |
| Chlachula J. | - Professor, Adam Mickiewicz University (Poland) |
| Redfern S.A.T. | - PhD, Professor, (Singapore) |
| Cheryomushkina V.A. | - Doctor of Biological Sciences, Professor (Russia) |
| Bazarnova N. G. | - Doctor Chemical Sciences, Professor (Russia) |
| Mohamed Othman | - Dr. Professor (Malaysia) |
| Sherzod Turaev | - Dr. Associate Professor (United Arab Emirates) |

Editorial address: 8, Kabanbay Batyr avenue, of.316, Nur-Sultan,
Kazakhstan, 010000
Tel.: (7172) 24-18-52 (ext. 316)
E-mail: natural-sciences@aiu.kz

International Science Reviews NST - 76153

International Science Reviews

Natural Sciences and Technologies series

Owner: Astana International University

Periodicity: quarterly

Circulation: 500 copies

CONTENT

Мұсабаева Б.Х., Романова М.А., Сабитова А.Н., Шарипхан Ж. СЕМЕЙ ӨНІРІНІҢ ДӘРІЛІК ӨСІМДІКТЕРІ ҚҰРАМЫНДАҒЫ БИОЛОГИЯЛЫҚ АКТИВТІ ЗАТТАР.....	5
Г.Ш. Омаркулова, Абилова П.Н., Тасболатұлы Н., Оңталапұлы С. ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ ИНЖЕНЕРИИ.....	13
С.С.Шубаев ВОЗМОЖНОСТИ PANDAS ДЛЯ ВИЗУАЛИЗАЦИИ ДАННЫХ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ.....	20
Жанабекова А.Ж. ЖЕРГІЛІКТІ ЖЕЛІНІ ПАЙДАЛАНА ОТЫРЫП, НАУҚАСТЫҢ АРНАЙЫ ДЕНСАУЛЫҚ КӨРСЕТКІШТЕРІН ҚАШЫҚТЫҚТАН БАҚЫЛАУ.....	28
Әбдіхалық Е.ӘЛЕМДІК ОРМАН ӨРТІ АЛДЫН – АЛАУ ШАРАЛАРЫНЫҢ АРТЫҚШЫЛЫҚТАРЫ МЕН КЕМШІЛІКТЕРІ ЖӘНЕ ОРМАН ӨРТІНІҢ ЖЫЛДАМДЫҒЫН ӨЗГЕРУ АУҚЫМЫ.....	35
Мухитова А. RFID АРҚЫЛЫ ЕСІККЕ КІРУДІ БАСҚАРУ.....	41
Кожанова Г. Г. АҚЫЛДЫ ЕСІК ҚОҢЫРАУЛАР ЖҮЙЕСІНІҢ ҚОЛДАНЫЛУ АЯСЫНА ҚАТЫСТЫ ӨЗІРЛЕУ.....	47
Құлмамұров С.А., Сансызбай Ә. С. ИНТЕЛЛЕКТУАЛДЫ БАСҚАРУ ЖҮЙЕЛЕРІНІҢ АППАРАТТЫҚ-БАҒДАРЛАМАЛЫҚ БӨЛІГІНЕ ҚОЙЫЛАТЫН ТАЛАПТАР.....	53

ВОЗМОЖНОСТИ PANDAS ДЛЯ ВИЗУАЛИЗАЦИИ ДАННЫХ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Шубаев С.С.

Международный университет Астана, Нур-Султан, Казахстан

diarys@mail.ru

Аннотация: Изучен способ визуализации данных информационной безопасности на языке Python, использованы библиотеки Pandas, Matplotlib. Установлена среда разработки программного кода Anaconda, Jupyter. Установлен в качестве тестовой среды сервер в Linux Ubuntu, на котором установлен Python 3, Pandas, Matplotlib, настроен доступ через SSH с клиентской рабочей станции. Были протестированы скрипты запросов на Pandas к источнику журнальных файлов в формате с расширением CSV, для их последующей визуализации. Изучены API средства JSON и способы для передачи из-под Java приложений данных в среду Python в датафреймы Pandas.

Ключевые слова: Pandas, Matplotlib, визуализация данных информационной безопасности, Elasticsearch, ELK, SIEM.

ВВЕДЕНИЕ

В части визуализации данных была запущена и испытана работа на языке Python, использована библиотека Pandas с датафреймами, библиотека визуализации Matplotlib, среда разработки использовалась Anaconda Jupyter – блокнот. Загрузка осуществлялась в виде формата файла с расширением csv.

Для визуализации данных информационной безопасности была взята выгрузка программы Wireshark.

Изучены стандарты файлов журналирования различных систем для работы по выборке визуализации данных по информационной безопасности.

Изучены протоколы передачи файлов журналирования такие как SNMP, NetFlow, Syslog.

Проведена экспериментальная работа программного кода на сервере и рабочей станции, для визуализации данных информационной безопасности

ВИЗУАЛИЗАЦИЯ В PANDAS

В этой статье мы хотим продемонстрировать еще одну возможность по визуализации данных в информационной безопасности с помощью программирования выборки из сетевого трафика программы Wireshark. Здесь будет рассмотрена библиотека Pandas для Python которая будет использовать библиотеку Matplotlib. С помощью Pandas можно будет обращаться к данным в виде запросов, которые в итоге можно будет визуализировать. Библиотеку Pandas

можно будет использовать для получения необходимой аналитики практически из любого аппаратного или программного устройства, достаточно чтобы имелись различные файлы журналов будь то с расширением CSV, либо их можно будет преобразовать в нужный нам формат данных так называемых датафреймы с помощью API JSON.

Эта статья служит для демонстрации больших возможностей Pandas для того, чтобы получить отличную аналитику по данным. В будущем можно будет использовать Pandas в качестве программирования и получения более точных данных для визуализации из централизованного хранилища данных таких как SIEM системы и как стека Elasticsearch. В стеке Elasticsearch уже имеются готовые решения в виде скриптов и дашбордов для визуализации данных, но тем не менее если нам понадобится видоизменить запросы получить более точно настроенную аналитику в разрезе, то программные возможности Pandas очень широки для этих целей.

Программа на Pandas была развернута в программной среде Jupyter в менеджере пакетов Anaconda, версия Python 3.7

Ниже приведен пример кода программы на Pandas для этого были выгружены сетевые данные из программы Wireshark. В качестве примера и демонстрации возможностей Pandas был записан сетевой трафик компьютера с помощью Wireshark за короткий период 30 минут – 1 час. Затем файл журнала был импортирован в файл с расширением csv, для дальнейшей передачи в Pandas. Далее в Pandas этот файл был передан в датафрейм Pandas для осуществления работы над данными смотреть рис.1

```

In [1]: import pandas as pd

In [4]: df = pd.read_csv('15_01_22_1.csv')

In [5]: df.head()

Out[5]:
   No.  Time  Source  Destination  Protocol  Info  Dest Port  Source Port  TCP Segment Len  Total Length  Packet length  Time relative  Time format
0    1  0.000000  192.168.71.125  213.180.193.234  TCP  1055 > 443 [ACK] Seq=1 443.0 1055.0 1.0 41.0 55 0.000000 0.000000
1    2  0.000000  192.168.71.125  149.154.167.99  TLSv1.2  Application Data 443.0 8434.0 306.0 346.0 360 0.000000 0.000000
2    3  0.027052  108.139.248.254  192.168.71.125  TCP  (TCP segment of a reassembled PDU) 1091.0 443.0 1400.0 1440.0 1454 0.027052 0.027052
3    4  0.027103  192.168.71.125  108.139.248.254  TCP  1091 > 443 [ACK] Seq=1 443.0 1091.0 0.0 76.0 90 0.027103 0.027103
4    5  0.038756  170.114.10.172  192.168.71.125  TLSv1.2  Change Cipher Spec: Encrypted Handshake Message 1111.0 443.0 81.0 81.0 105 0.038756 0.038756

In [6]: df.shape
Out[6]: (14860, 13)

In [7]: df_r = df[df.Source=='192.168.71.125']

In [8]: df_r.shape
Out[8]: (7611, 13)

```

Рисунок 1 – Датафрейм Pandas

Вначале импортирую библиотеку Pandas, загружаю файл csv в датафрейм *df*. Выводим датафрейм на экран в блокнот Jupyter. Хотим получить информацию по количеству строк и столбцов делаем запрос *df.shape* который выводит количество строк и столбцов.

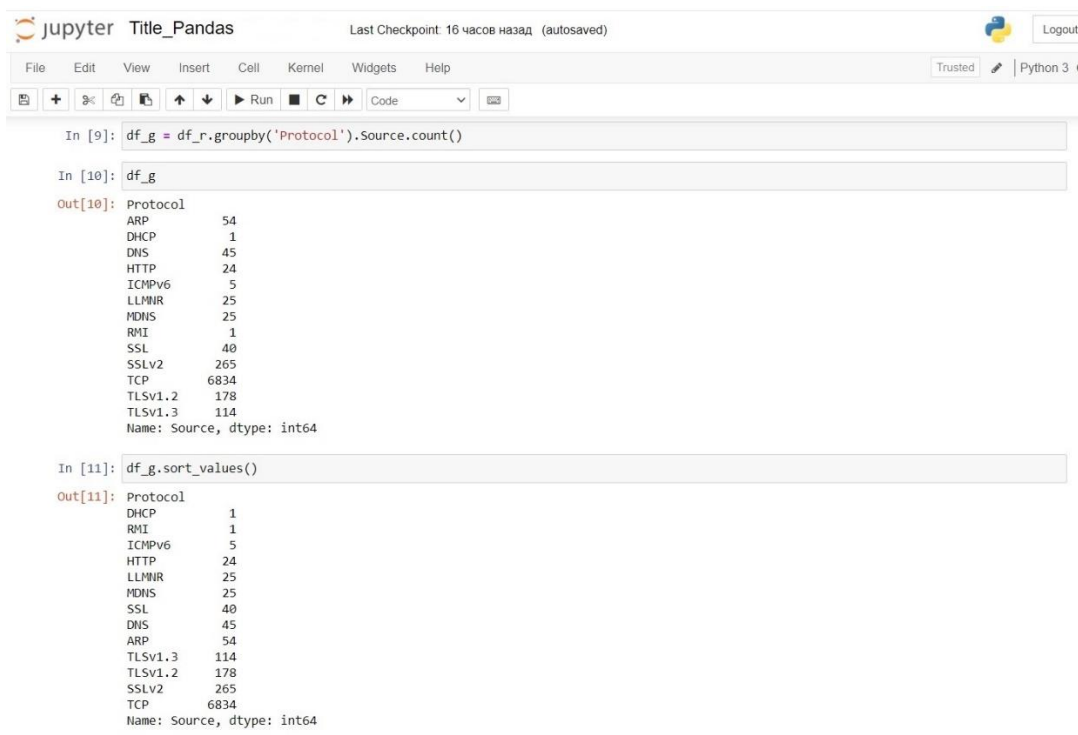
Так как нас интересует входящий трафик ввожу запрос в котором прописываю свой IP-address компьютера который входит в исключения и который не является источником для правил запросов –

```
df_r = df[df.Source!='192.168.71.125'].
```

В ответ выводится количество 7611 строк.

Далее `groupby('Protocol')` и `count()`, выводит пакеты для каждого протокола, так как каждая строка является пакетом (Рис.2).

Вывожу по порядку возрастания `sort_values`



```

In [9]: df_g = df_r.groupby('Protocol').Source.count()

In [10]: df_g
Out[10]: Protocol
ARP          54
DHCP          1
DNS          45
HTTP         24
ICMPv6        5
LLMNR        25
MDNS         25
RMI           1
SSL          40
SSLv2       265
TCP       6834
TLSv1.2      178
TLSv1.3      114
Name: Source, dtype: int64

In [11]: df_g.sort_values()
Out[11]: Protocol
DHCP          1
RMI           1
ICMPv6        5
HTTP         24
LLMNR        25
MDNS         25
SSL          40
DNS          45
ARP          54
TLSv1.3      114
TLSv1.2      178
SSLv2       265
TCP       6834
Name: Source, dtype: int64

```

Рисунок 2 – Фрагмент кода в Pandas

Далее производится замена имен столбцов с пробелами на () нижнее подчеркивание для дальнейших запросов по именам столбцов. Рис. 3

```
%matplotlib inline
```

```

df_r.rename(columns={"Dest Port":"Dest_Port",
"Source port":"Source_port",
"TCP Segment Len":"TCP_Segment_Len",
"Total Length":"Total_Length",
"Packet length":"Packet_length",
"Time relative":"Time_relative",
"Time format":"Time_format" }, inplace = True)

```

```
df_r.head()
```

```
df_r[df_r['Protocol']=='TCP'].head()
```

```
df_r[df_r['Protocol']=='TCP'].Total_Length.hist(bins=15)
```

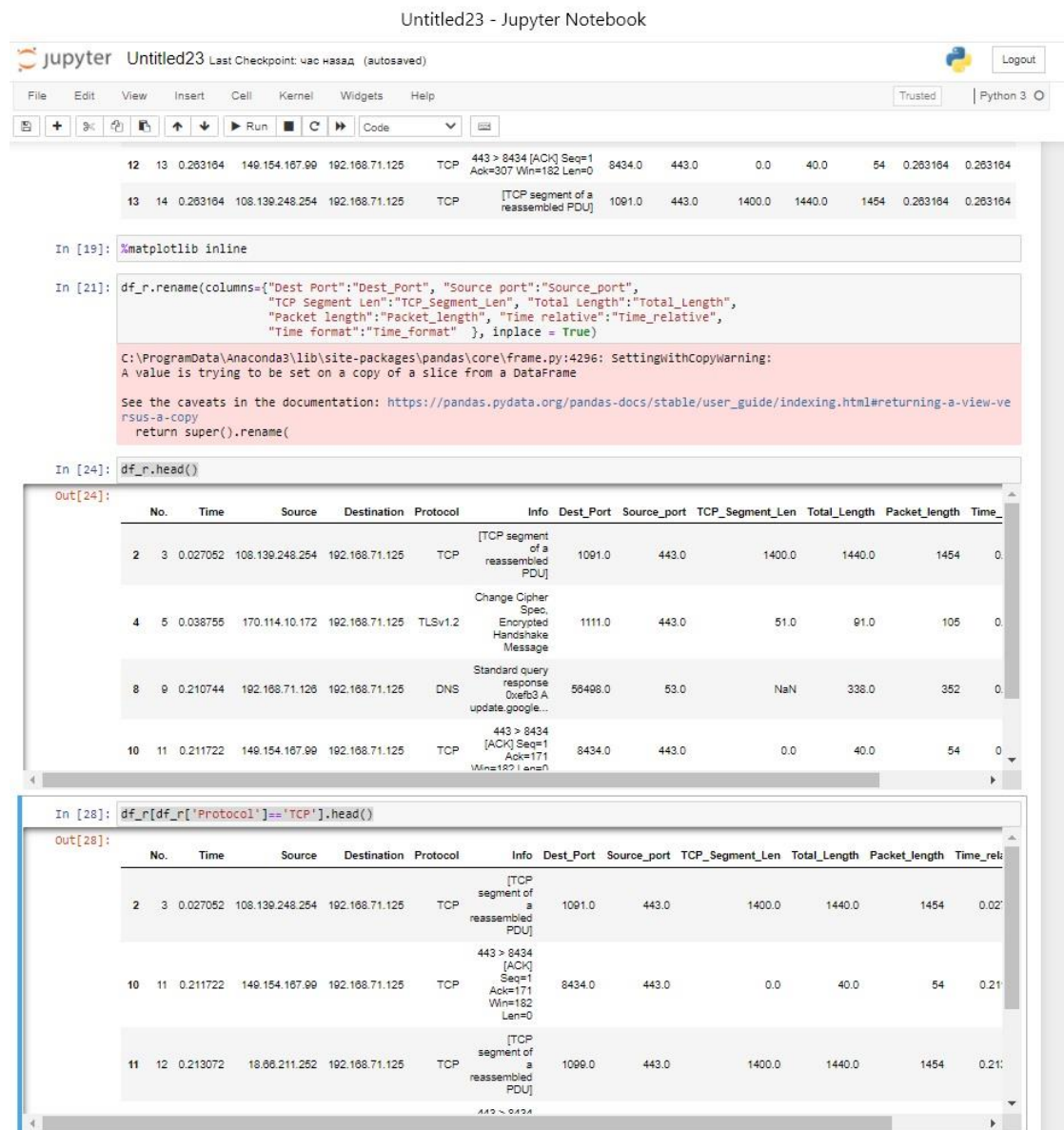


Рисунок 3 – Запросы

Выводим гистограмму для tcp-пакетов total_length и HTTP-пакетов total_length. На графике мы можем увидеть, что большинство пакетов имеют размер от 1400 до 1500 байт смотреть Рис.4

```
df_r[df_r['Protocol']=='HTTP'].Total_Length.hist(bins=15)
```

Рис.4

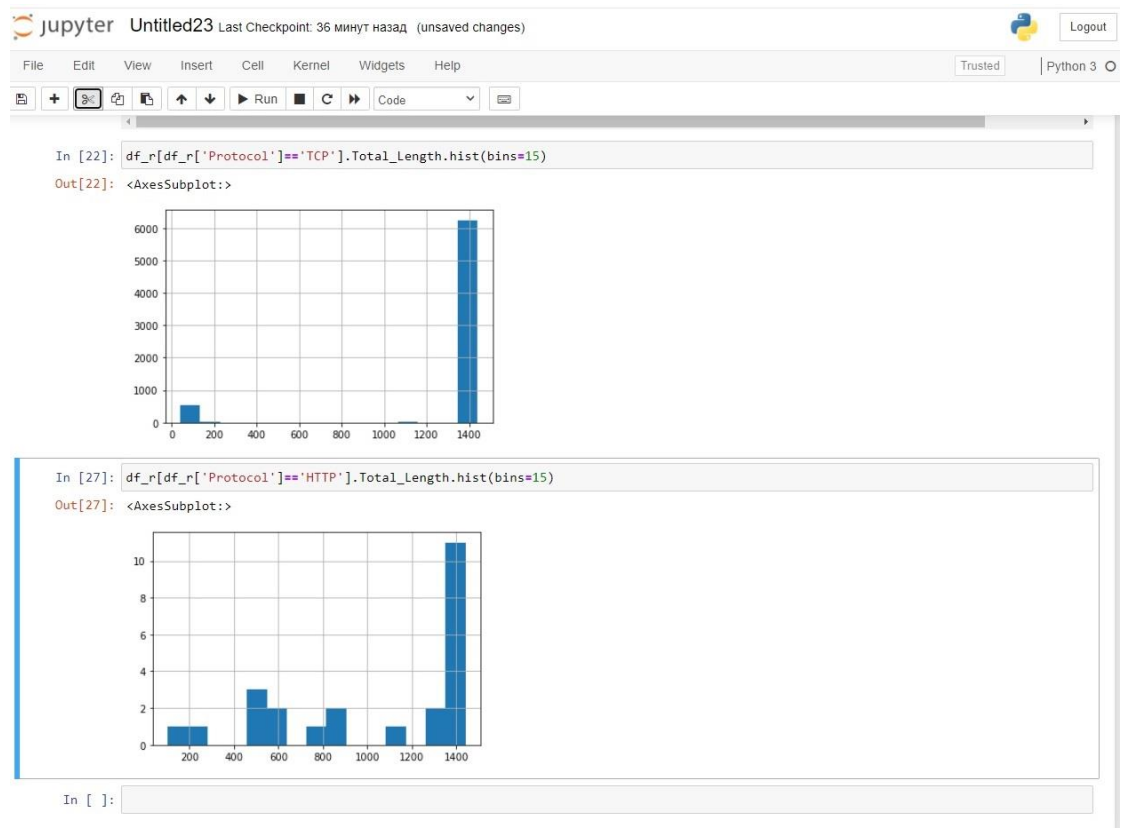


Рисунок 4 – Размеры пакетов

Теперь необходимо вычислить сумму `total_length` для каждого протокола и вывести в виде гистограммы. Для того чтобы преобразовать байты в мегабайты необходимо разделить $1024/1024$ смотрим рис. 5

```
df_s=df_r.groupby('Protocol').Total_Length.sum()
df_s_mb=df_s/(1024*1024)
df_s_mb.plot(kind='bar')
```

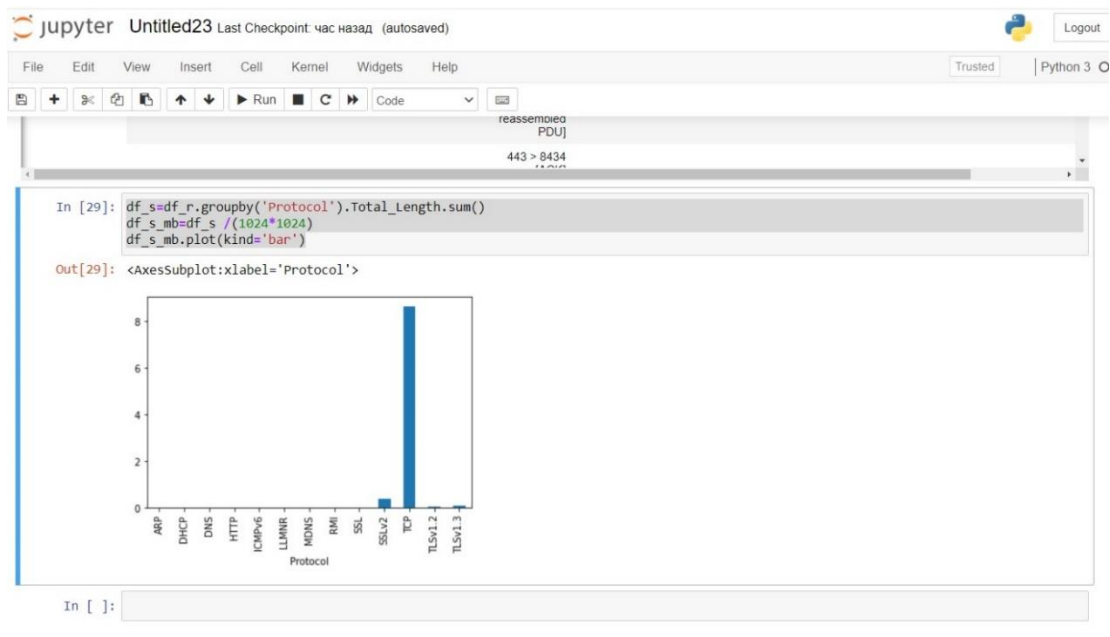


Рисунок 5 – Преобразование

Далее необходимо вывести на печать количество пакетов по протоколу
смотреть рис. 6

```
df_p=df_r.groupby('Protocol').Source.count()
df_p.plot(kind='bar')
```

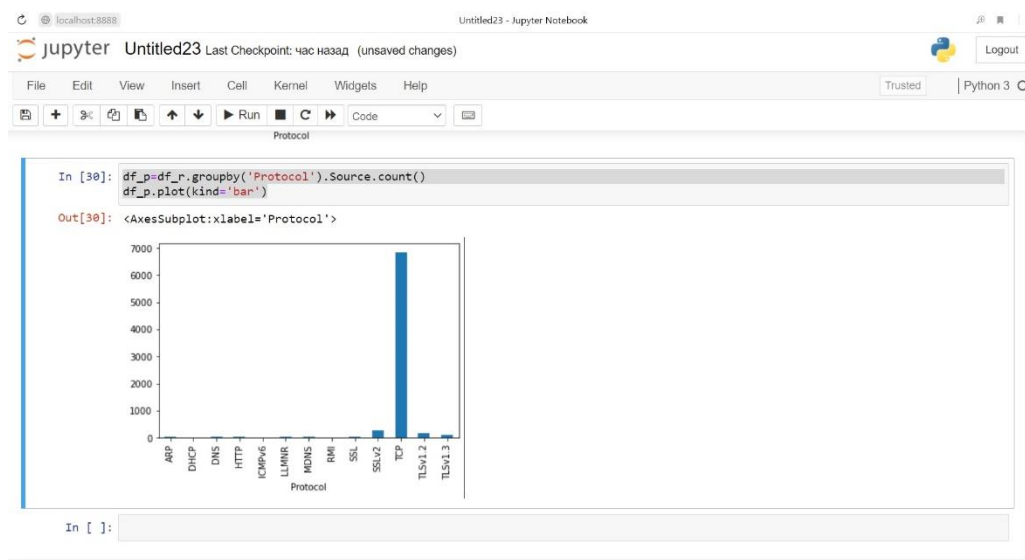


Рисунок 6 – Количество пакетов по протоколу

Выводим общий размер пакета на количество пакетов, чтобы найти средний
размер пакета.

```
df_a_p=df_s / df_p
```

```
df_a_p.plot(kind='bar')
```

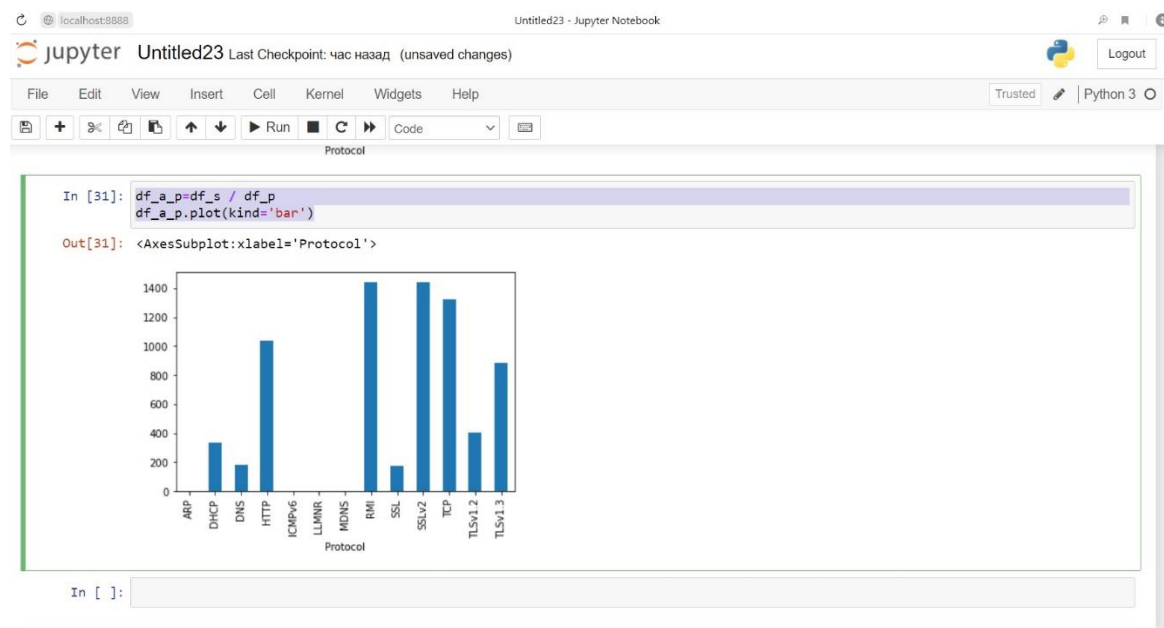


Рисунок 7 - Результат

В данном графике отображаются кадры ARP хотя размер равен нулю, так как мы вычисляем total_length пакетов ip, в то время как кадры arp относятся ко 2-му уровню модели OSI. Если бы была необходимость их включить, нужно было бы принять во внимание размер кадра Ethernet, вместо размера IP-пакета

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ:

1. Python. Библиотека Pandas, часть 1
<https://istories.media/workshops/2021/03/05/python-biblioteka-pandas-chast-1/>
2. Python. Библиотека Pandas, часть 2
<https://istories.media/workshops/2021/03/12/python-biblioteka-pandas-chast-2/>
3. Python. Библиотека Pandas, часть 3
<https://istories.media/workshops/2021/03/19/python-biblioteka-pandas-chast-3/>
4. Community tutorials https://pandas.pydata.org/pandas-docs/stable/getting_started/tutorials.html
5. Pandas. Работа с данными. 2-е изд. - devpractice.ru. 2020. - 170 с.: Абдрахманов М.И.
6. Learning Pandas. Michael Heydt
7. Введение в pandas: анализ данных на Python <https://khashtamov.com/ru/pandas-introduction/>
8. Python Data Science Handbook by Jake VanderPlas O'Reilly
9. Getting started with Python for science
<https://scipy-lectures.org/intro/matplotlib/index.html>
10. Интерактивная визуализация данных при помощи Plotly: строим красивые графики с Express и Cufflinks <https://habr.com/ru/company/skillfactory/blog/506974/>
11. Визуализация данных в Python с помощью Plotly
<https://stepik.org/lesson/360195/step/1?unit=345176>
12. Running a notebook server https://jupyter-notebook.readthedocs.io/en/latest/public_server.html