ОБНАРУЖЕНИЕ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ НА ОСНОВЕ ТЕХНОЛОГИИ НЕЙРОННЫХ СЕТЕЙ

Букин А.В.[1], Самонов А.В.[2], Тихонов Э.И.[3]

Цель исследования: *разработать модельное, алгоритмическое и программное обеспечение для обнаружения в режиме реального времени попыток нарушения корректного функционирования систем критической информационной инфраструктуры.*

Метод исследования: *анализ современных методов машинного обучения и нейросетевых технологий, синтез и моделирование корректного поведения программ, алгоритмизация процессов обучения и применения нейросетей, экспериментальные исследования разработанных алгоритмов и программ на стенде.*

Результат исследования: *дана характеристика методов машинного обучения и нейросетевых технологий, используемых для обнаружения программно-технических воздействий и инцидентов информационной безопасности. Разработан метод решения данной задачи на основе нейросетей с LSTM и FFN архитектурами. Дано описание алгоритма и фрагментов программной реализации метода на языках программирования Python3 и Go с использованием библиотек Tensorflow и Keras. Важным достоинством предложенного подхода является наличие возможности адаптации нейросети в случае изменения режима и условий функционирования системы. Полученные в ходе экспериментов результаты свидетельствуют о возможности и целесообразности применения данного подхода для обнаружения программно-технических воздействий на критические системы информационной инфраструктуры в масштабе времени близком к реальному с высоким уровнем достоверности.*

Научная новизна: *состоит в применении технологии глубокого обучения на основе долгой краткосрочной нейросети LSTM, обладающей способностью адаптации к изменяющимся режимам и условиям, для решения задачи обнаружения признаков нарушения корректного функционирования узлов информационно-телекоммуникационных систем в режиме реального времени.*

Ключевые слова: *временной ряд, глубокое обучение, методы машинного обучения, обнаружение аномалий, рекуррентные нейросети, системы обнаружения вторжений, функция потерь*

DOI:10.21681/2311-3456-2022-5-61-73

Введение

Как отмечено в отчете, представленном отечественной компанией Positive Technologies[4], в настоящее время наблюдается беспрецедентное по своему размаху и интенсивности увеличение кибератак на российские информационные ресурсы. Наибольшее распространение при этом получили DDoS-атаки, взлом сайтов крупных компаний и популярных ресурсов, существенно возросло количество целенаправленных атак на государственный сектор, банковскую сферу, топливно-энергетический комплекс, научные институты и организации, связанные с оборонно-промышленным комплексом.

В материале, опубликованном американской компанией Check Point[5] сказано, что 2022 год начался с массированной эксплуатации одной из самых серьезных уязвимостей в Интернете – Apache log4j и продолжился полномасштабной кибервойной с началом российской специальной военной операции. Отмечено,

4 https://www.ptsecurity.com/upload/corporate/ru-ru/analytics/positive-research-2022-rus.pdf

5 https://blog.checkpoint.com/2022/07/26

1 Букин Александр Вячеславович, научный сотрудник Военно-космической академии им. А.Ф. Можайского, Санкт-Петербург, Россия. E-mail: bukinav.mail@gmail.com

2 Самонов Александр Валерьянович, кандидат технических наук, доцент, старший научный сотрудник, Военно-космической академии имени А.Ф. Можайского, Санкт-Петербург, Россия E-mail: a.samonov@mail.ru ORCID: 0000-0002-0390-4481

3 Тихонов Эдуард Игоревич, кандидат технических наук, старший научный сотрудник Военно-космической академии им. А.Ф. Можайского, Санкт-Петербург, Россия, E-mail: inta.et@gmail.com

что основными целями атак являются сектор образования и исследований, рост составил 53% по сравнению со 2 кварталом 2021 года, в среднем более 2,3 тыс. атак на организацию каждую неделю. За ними следует правительственный и военный сектор, где в среднем в неделю совершалось 1,6 тыс. нападений, что на 44% больше по сравнению с аналогичным периодом прошлого года. Далее идут сектора интернет-провайдеров и MSP, здравоохранения и связи, где в среднем на организацию приходится 1,3 тыс. атак в неделю, что представляет собой двузначный рост по сравнению с прошлым годом.

В публикациях Европейского агентства по кибербезопасности (European Union Agency for Cybersecurity, ENISA)[6] отмечено, что все более активное участие в разработке и применении кибероружия будут принимать правительственные организации и спецслужбы, а также, что распределенные атаки на отказ в обслуживании (DDoS) будут более целенаправленными, более настойчивыми и многовекторными.

Постоянное совершенствование технологий и средств реализации атак в отношении критических систем информационной инфраструктуры (КСИИ) обуславливает настоятельную необходимость создания адекватных или превосходящих методов и средств защиты от них. Для защиты КСИИ от программно-технических воздействий (ПТВ) в настоящее время используются межсетевые экраны (МЭ), системы обнаружения и предупреждения вторжений (СОПВ, IDS/IPS), системы предотвращения потери данных (DLP, Data Loss Prevention), системы управления событиями информационной безопасности (SIEM, Security information and event management), антивирусы и др.

Современные методы обнаружения ПТВ, приводящих к инцидентам информационной безопасности (ИИБ), можно разделить на две основные категории: распознавание злоупотреблений и выявление аномалий. Методы распознавания злоупотреблений, описываемых с помощью сигнатур известных атак, имеют высокую точность и низкий уровень ложных срабатываний, но неспособны обнаруживать атаки, для которых отсутствуют сигнатуры. Методы обнаружения аномалий позволяют выявлять ранее неизвестные атаки, но имеют высокий уровень ложных срабатываний.

В статье представлен краткий анализ методов машинного обучения и нейросетевых технологий, используемых для обнаружения аномалий. Предложен метод решения данной задачи на основе нейросетей с LSTM и FFN архитектурами. Дано описание алгоритма и фрагментов программной реализации данного метода. Полученные в ходе экспериментов результаты свидетельствуют о возможности и целесообразности применения данного подхода для обнаружения ПТВ на КСИИ в масштабе времени близком к реальному с высоким уровнем достоверности.

1. Анализ методов машинного обучения и нейросетей, используемых для обнаружения аномалий в системах кибербезопасности

В настоящее время технологии машинного обучения и нейросетей применяются для решения множества задач классификации, прогнозирования и принятия решений. В системах киберзащиты эти технологии используются для выявления закономерностей в поведении систем, обнаружения аномального поведения и противодействия компьютерным атакам. Описание примеров и способов применения нейросетевых технологий для обнаружения событий информационной безопасности представлено в целом ряде публикаций и работ.

В первую очередь отметим публикации [1 – 6], в которых представлены обзорные аналитические материалы о способах применения методов машинного обучения и нейросетевых технологий для обеспечения кибербезопасности. Так в статьях [1, 2] представлен анализ современных технологий обнаружения компьютерных атак, которые можно разделить на две основные категории: распознавание образов (pattern recognition) и выявление аномалий (anomaly detection). Отмечено, что методы распознавания образов, основанные на сигнатурном анализе, имеют высокую точность и низкий уровень ложных срабатываний, но неспособны обнаруживать атаки, для которых отсутствуют сигнатуры. Методы обнаружения аномалий позволяют выявлять ранее неизвестные атаки, но имеют высокий уровень ложных срабатываний, обусловленных сложностью разработки профилей нормального поведения контролируемых систем. Оптимальным решением является совместное использование этих технологий, учитывая тот факт, что результаты, полученные с помощью методов обнаружения аномалий, могут использоваться для разработки новых сигнатур.

В статье [2] отмечено, что самым большим преимуществом подхода, основанного на аномалиях, является его способность обнаруживать атаки нулевого дня, поскольку он не зависит от используемой базы

6 https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends

Рис.1. Классификация методов глубокого обучения, используемых для обнаружения вторжений

данных сигнатур, а позволяет выявлять отклонения от нормального поведения. Поведение каждой целевой системы уникально, поэтому подходы, основанные на обнаружении аномалий, должны использовать индивидуальные профили, которые, в свою очередь, затрудняют злоумышленнику точное определение того, какие действия он может выполнить, не вызывая тревоги. Отмечено, что недостатками систем обнаружения атак, основанных на обнаружении аномалий, являются: высокий уровень ложноположительных результатов и необходимость формирования профилей нормального поведения контролируемой системы.

В статье [3] представлен детальный анализ таких архитектур нейронных сетей как неокогнитрон, автокодировщики, сверточные нейронные сети, ограниченная машина Больцмана, глубокие сети доверия, сети долго-краткосрочной памяти, управляемые рекуррентные нейронные сети и сети остаточного обучения. Дана характеристика программных библиотек, реализующих методы глубокого обучения.

В статье [4] приведена классификация и краткая характеристика методов глубокого обучения, используемых для обнаружения вторжений посредством выявления аномалий (рис.1). В статьях [5 – 7] представлен сравнительный анализ достоинств и недостатков традиционных методов машинного обучения (machine learning, ML) и технологий глубокого обучения (deep learning, DL). Определено, что технологии DL целесообразно применять на больших объемах данных для неконтролируемого или полууправляемого изучения и установления связей и закономерностей в процессах и событиях. В статье обоснованы следующие характеристики качества методов обнаружения атак: точность (precision), чувствительность (sensitivity), среднее гармоническое значение точности и отзыва (F1-score), кривая ROC (receiver operating characteristic), описывающая компромисс классификатора между правильно-положительными (true positive) и ложно-положительными (false positive) решениями, характеристика производительности AUC (area under curve ROC). Приведены характеристики точности ML и DL алгоритмов по обнаружению атак из базы данных NSL-KDD ‘99.

В ряде источников предлагается для обнаружения аномалий по неизвестным данным использовать генеративно-состязательные сети (Generative Adversarial Networks, GAN). Так в [8] данная модель применялась для экспериментального исследования набора данных ботнетов ISCX. В [9] представлен анализ производительности сетей данного типа. В [10] для обнаружения аномалий применяется модель на основе двунаправленной GAN (BiGAN), которая дополнительно проводит обратное отображение реальных данных в скрытое пространство. Помимо экономии времени, BiGAN способствует более эффективному извлечению признаков сетевого трафика. Данная модель на наборе данных KDD Cup 99 показала точность 93,24%.

Многие авторы публикаций отмечают, что для одним из наиболее перспективных подходов к обна-

Таблица 1

Уровень опасности для КСИИ обнаруженного ПТВ

Значение	Уровень опасности	Описание
1	Не принимается в расчет	кратковременное нарушение в работе информационной инфраструктуры, не влияющее на процесс передачи данных и обработки данных, вызванное единичными сбоями
2	Допустимый	нарушения, носящие единичный характер и не взаимосвязанные друг с другом
3	Нежелательный	нарушения, значительно влияющие на процесс передачи или обработки данных, вызванные сбоями
4	Недопустимый	преднамеренные (в том числе циклические) ПТВ на КСИИ, приводящие к ИИБ

ружению кибератак в режиме времени близком к реальному является использование рекуррентных нейронных сетей (Recurrent Neural Network, RNN) [5, 11 – 21]. Отличительной особенностью RNN является наличие обратной связи, что позволяет анализировать последовательные данные, такие как временные ряды. Анализируя последовательность измерений различных параметров текущего процесса, нейросеть обучается предсказывать его состояние в следующий момент времени. Если предсказанное RNN состояние отличается от текущего, регистрируется аномалия. В [11] RNN применяется для бинарной и мультиклассовой классификации наборов сетевых данных NSL-KDD. Недостатком стандартных RNN являются проблемы с исчезновением градиента и нехватка памяти для использования информации за предыдущие моменты времени. В [12] рассмотрены методы обнаружения вторжений, основанный на долгой кратковременной памяти (Long Short-Term Memory, LSTM), которые позволяют справляться с данными проблемами посредством увеличения количества обучающих параметров. В статье [13] используется RNN с управляемым рекуррентным блоком (Gated Recurrent Units, GRU), которая требует меньше параметров для обучения. Эксперименты на наборе данных KDD Cup 99 показывают точность обнаружения атак, равную 99,91%. В статьях [14 – 19] представлены примеры использования методов глубокого обучения для классификации и обнаружения атак на информационно-телекоммуникационные системы.

Анализ публикаций показал, что одним из наиболее перспективных подходов к обнаружению аномального поведения контролируемых КСИИ в режиме реального времени является применение методов глубокого обучения, в частности, рекуррентных сетей LSTM. Основными проблемными являются вопросы, связанные с построением профилей нормального поведения контролируемых систем, определение параметров и настройка нейросети, обеспечение ее адаптируемости к изменяющимся условиям.

2. Алгоритм обнаружения аномального поведения систем на основе технологии нейросетей

В данной статье представлен алгоритм обнаружения событий информационной безопасности (СИБ), обусловленных программно-техническими возможностями (ПТВ) на КСИИ, с помощью нейросетей с LTSM и FFN архитектурами. Эти нейросети обеспечивают решение задачи регрессии, т.е. прогнозирования значений параметров, характеризующих уровень опасности ПТВ для атакуемой системы. Разработанный на их основе алгоритм обеспечивает обнаружение аномалий и ранних признаков ПТВ, которые могут привести к инцидентам информационной безопасности (ИИБ). Под ПТВ понимается целенаправленное аппаратно-программное или программное воздействие, а также их комбинация на КСИИ, приводящие к нарушению процесса их функционирования. Посредством нейросети формируется значение выходной переменной, которое характеризует уровень или степень опасности для КСИИ обнаруженного ПТВ (табл. 1).

Схема алгоритма обнаружения аномального поведения систем, обусловленных ПТВ, на основе технологии нейросетей представлена на рис. 2. Основными этапами являются:

— определение набора входных переменных нейросети, выход числовых значений которых за установленные пределы трактуется как признаки ПТВ и СИБ;

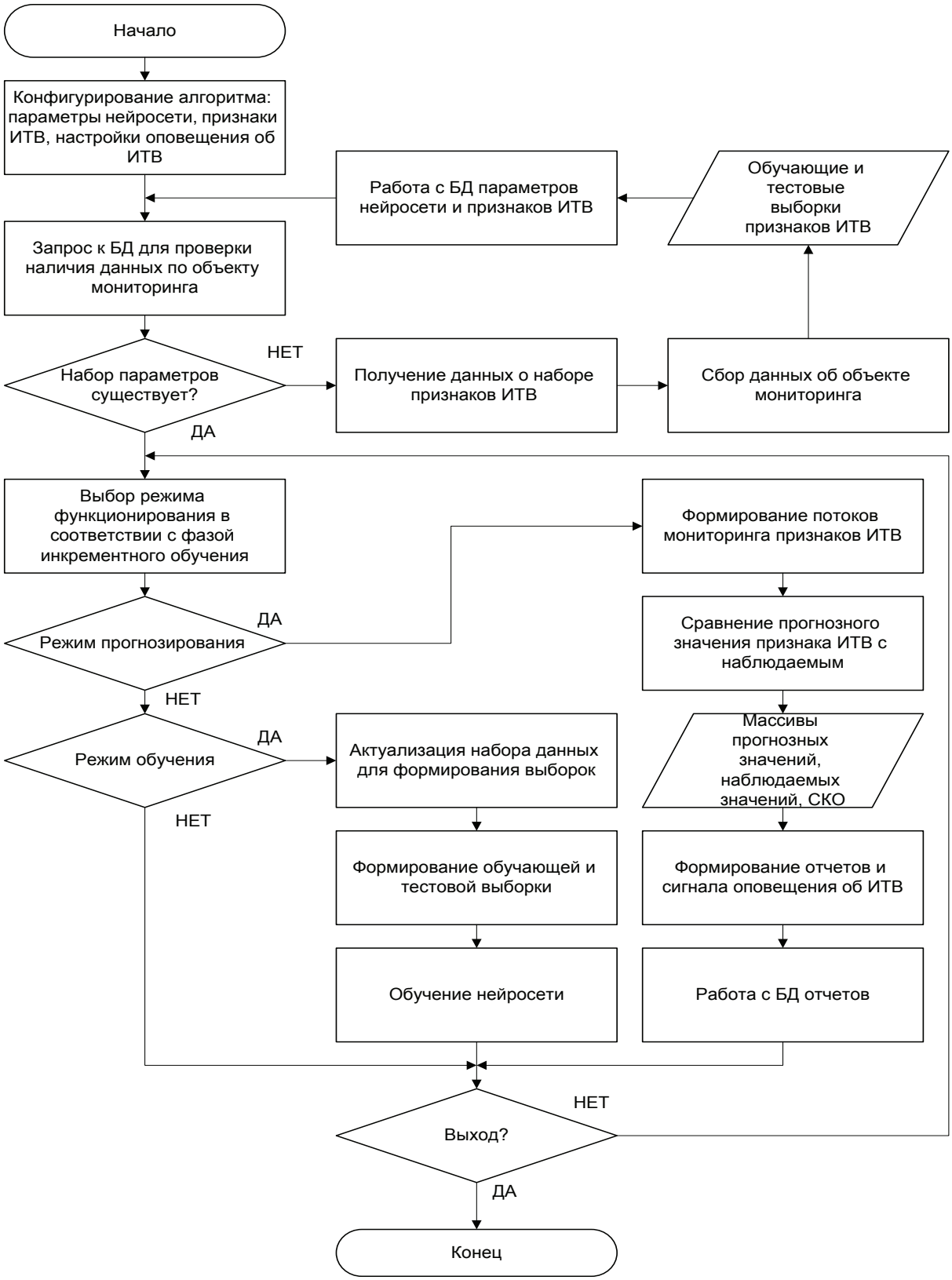

Рис. 2. Схема алгоритма выявления СИБ и обнаружения ИТВ с использованием технологии нейросети

DOI:10.21681/2311-3456-2022-5-61-73

— сбор и получение данных о контролируемом объекте;
— обучение нейросети;
— прогнозирование и обнаружение аномального поведения.

Работоспособность алгоритма выявления аномалий зависит от корректности выбора набора входных переменных нейросети, выход числовых значений которых за установленные пределы трактуется как признаки ПТВ и СИБ. Алгоритм предназначен для обнаружения аномалий, вызванных ПТВ на узел информационной инфраструктуры, поддерживающий сетевые сервисы: сервер QoS, сервер прикладных служб HTTP, FTP, SMTP, SNMP и др. Соответственно, в качестве признаков могут использоваться: количество активных процессов; данные об активных (новых, удаленных, измененных) учетных записях пользователей; данные о сетевых соединениях и запросах; изменения в записях системных планировщиков; данные о процессах и демонах (запуск, останов); степень загруженности процессора; данные о использовании памяти. Аномальные изменения значений данных признаков сигнализируют о несанкционированной активности пользователей, ПТВ, функционировании вредоносного программного обеспечения (ВПО). Значения признаков могут быть получены с использованием сторонних программных средств, например – кроссплатформенной среды *osquery* (https://github.com/osquery). Числовые значения и набор независимых переменных (признаков) конкретизируются при отработке алгоритмов на макете. Также на макете отрабатывается и конкретизируется количество и диапазоны значений выходных зависимых переменных.

На этапе конфигурирования алгоритма формируется набор параметров, необходимых для его успешного функционирования с учетом особенностей объекта, подлежащего анализу. К основным параметрам относятся:

— временной ряд, характеризующий признаки ПТВ (один или несколько);
— граничные значения выходной переменной, отвечающей за формирование оповещения об ПТВ или СИБ;
— варианты функции потерь (loss function);
— процедуры оптимизации (optimization procedure). В начале экспериментальной отработки на стенде целесообразно использовать стандартную функцию потерь, представляющую собой сумму квадратичных ошибок (sum of squared errors, SSE). Для оптимизации используются оптимизаторы RMSProp и Adam (Adaptive Moment Estimation) из пакета Keras. В процессе конфигурирования формируется структура нейросети. Данная операция выполняется в соответствии с последовательностью, принятой для используемых в нашем случае библиотек Tensorflow и Keras [20, 21].

Библиотека Keras предоставляет возможность работы с несколькими типами RNN: слои классов keras.layers.SimpleRNN, keras.layers.GRU, keras.layers.LSTM. Для решения рассматриваемой в данной статье задачи обнаружения аномалий посредством прогнозирования временного ряда используются слои LSTM. Обучение LSTM сети осуществляется на тренировочном наборе, включающем один или несколько временных рядов в соответствии с количеством входных параметров нейросети, представленных в виде пар входных и выходных последовательностей.

В некоторых ситуациях возникает необходимость представления наборов данных для двух рядов независимых величин (два независимых параметра) и одной зависимой величины с необходимостью прогнозирования всех трех рядов. В этом случае модель LSTM модифицируется до многомерной модели LSTM для серий с несколькими входами (Multiple Input Series) или модели LSTM для нескольких параллельных серий (Multiple Parallel Series) [20]. В обоих случаях представления наборов данных входной слой сети преобразуется в соответствии с их размерностью:

LSTM(units=32, activation='relu', input_shape=(n_steps, n_features)

Во втором случае модифицируется и выходной полносвязный слой в соответствии с размерностью данных:

model.addELayer (Dense(n_features)),

n_features = X.shape[2] – число входных независимых рядов метрик;

n_steps = 100 – число элементов в n-грамме.

При формировании обучающего и тестового набора данных производится нормализация данных. Процедура нормализации на языке Python выглядит следующим образом:

```
def normalize(result):
  result_mean = result.mean() # вычисление
среднего значения набора данных
       result_std = result.std()         #
вычисление стандартного отклонения
    result -= result_mean
    result /= result_std
    return result, result_mean
```

Для проверки работоспособности алгоритма и корректности модели нейросети разработана программа обнаружения СИБ и ПТВ на языке Python3. В ходе экспериментов использовалась нейросеть с архитектурой «стековая LSTM» со следующей структурой:

- количество элементов в n-граммах – 100;
- количество слоев – 3;
- первый слой по количеству нейронов соответствует длине входной последовательности (*n-1*), коэффициент переобучения Dropout = 0.2;
- второй слой LSTM – 130 нейронов с коэффициентом переобучения = 0.2;
- третий слой LSTM – 100 нейронов;
- выходной слой, состоящий из одного нейрона, полносвязный (класс Densely-connected) с линейной функцией активации.

В результате экспериментов подтверждено, что данная структура нейросети в процессе работы алгоритма обеспечивает решение задачи прогнозирования и обнаружения аномалий для одного контролируемого параметра. Для обработки нескольких параметров необходимо изменить структуру нейросети в соответствии с приведенными выше рекомендациями. При проведении экспериментов в качестве исходных данных были использованы:

- временной ряд, содержащий данные о загруженности процессора в течение 660 секунд (данные получены из среды *osquery* с использованием Python3 bindings - *osquery-python)*;
- временной ряд, содержащий 660 значений, данные в котором соответствуют времени ответа сервера HTTP, формируемого по нестандартному закону распределения: в 2/3 случаев генерируются числа [0; 0.5) с уменьшающейся вероятностью [0.5; 1); аномальные данные соответствуют распределению Гаусса в диапазоне [0.9; 1).

Описание начальной структуры LSTM сети для реализации алгоритма на языке Python3 с использованием библиотек Tensorflow и Keras имеет следующий вид:

```
def generate_model():
    model = Sequential()
    # Первый слой соответствует длине входной последовательности
model.add(LSTM(input_shape=(sequence_length-1,1),
    units=32,return_sequences=True))
    model.add(Dropout(0.2)) # исключение переобучения
    # Второй слой LSTM из 128 нейронов
    model.add(LSTM(units=128,    return_sequences=True))
    model.add(Dropout(0.2))
    # Третий LSTM слой из 100 нейронов
     model.add(LSTM(units=100,    return_sequences=False))
    model.add(Dropout(0.2))
   # Полносвязный выходной слой с линейной ФА
    model.add(Dense(units=1))
    model.add(Activation('linear'))
   # алгоритм оптимизации, функция потерь
       model.compile(loss='mean_squared_error', optimizer='rmsprop')
    return model
```

Для формирования обучающего и тестового наборов предусмотрена функция *split_sequence*, входными параметрами для которой являются: массив данных, содержащий временной ряд параметра, точка начала тренировочного набора в массиве данных, точка окончания тренировочного набора, точка начала тестового набора, конечная точка тестового набора. Ниже приведен фрагмент функции, в котором представлены основные операции по формированию тренировочного и тестового наборов:

```
def  split_sequence(data,  train_start, train_end, test_start, test_end):
   result = []  # массив н-грамм
# диапазон можно задать и как длина дата минус sequence_length
   for index in range(train_start, train_end - sequence_length):
    result.append(data[index:  index  + sequence_length])
    result = np.array(result)
    result, result_mean = normalize(result)
    print("Размерность  массива  TH : ", result.shape)
   train = result[train_start:train_end, :]
    np.random.shuffle(train)
    X_train = train[:, :-1]
    y_train = train[:, -1]
    # Аналогично – формирование тестового набора
  …
    X_test = resultt[:, :-1]  # Отделяем выборки от меток, ВЫБОРКА
     y_test = resultt[:, -1]   # МЕТКИ
   …
 return X_wtrain, y_wtrain, X_wtest, y_wtest
```

В ходе проверки работоспособности модели сети в обоих случаях наборы формировались следующей командой, определяющей их размерность:

X_train, y_train, X_test, y_test = prepare_data(rez, 0, 600, 400, 660)

Для обучения использован метод инкрементного обучения, рекомендуемый для нейросетей с обучением с учителем. Метод позволяет реализовать не-

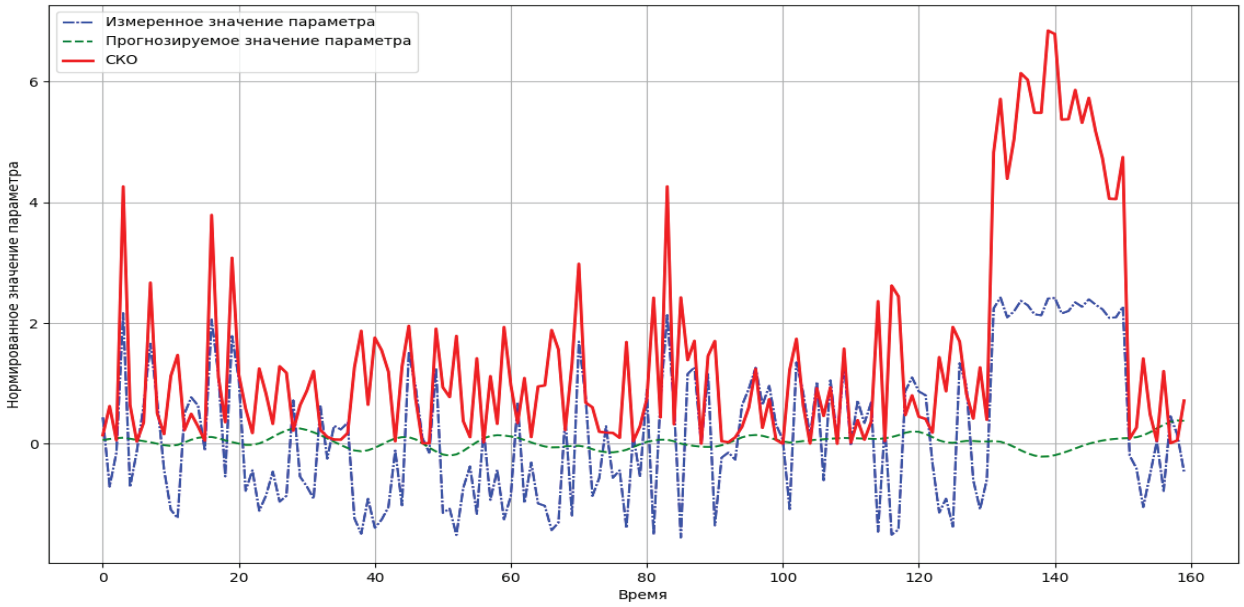

Рис. 3. Графики времени ответа сервера HTTP и СКО фактического значения параметра от прогнозируемого

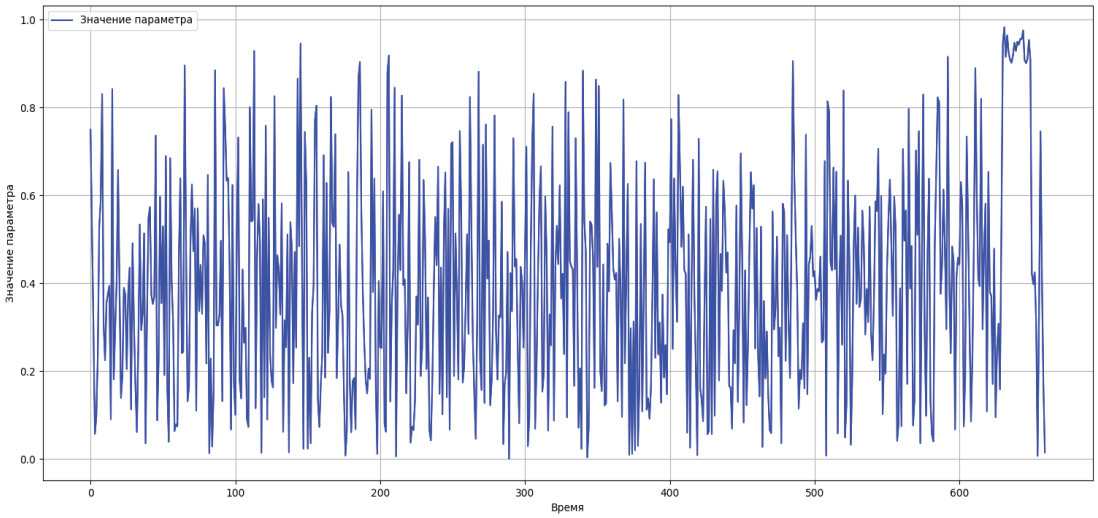

Рис. 4. Исходный набор данных для формирования тестовой и обучающей выборки (время ответа сервера HTTP, 660 измерений, 660 секунд)

прерывный процесс обработки данных в реальной системе обнаружения ПТВ путем чередования циклов прогнозирования (например, построение прогноза на 60минутный интервал через каждые 30 минут) и тренировки модели с помощью накопленных данных (используются данные за предыдущие 24 часа). Сравнение прогнозируемого значения параметра и его наблюдаемого значения, а также мониторинг статистической метрики – индикатора аномалии, проводятся ежеминутно. Необходимая временная последовательность уточняется и формируется по результатам отработки модели на стенде.

В результате работы программы формируются графики, отображающие изменение во времени значения контролируемой величины (рис.3). В качестве метрики-индикатора аномалии используется значение среднеквадратической ошибки (СКО), характеризующее степень отклонения измеренного фактического значе-

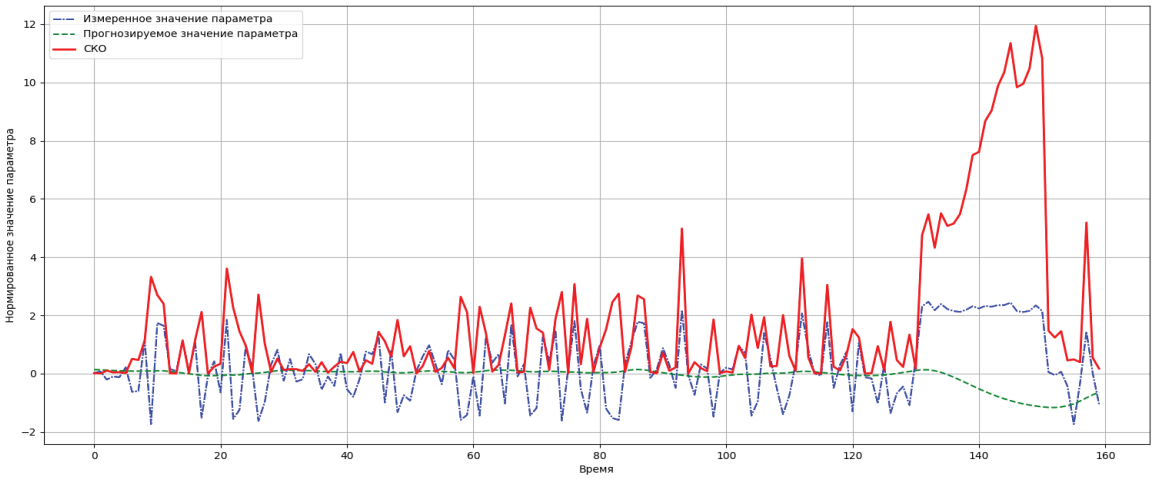

Рис. 5. Графики времени ответа сервера HTTP и СКО фактического значения параметра от прогнозируемого

ния параметра от прогнозируемого. Представленные графики позволяют визуально определить аномалию контролируемого параметра по выбросу значения метрики-индикатора, которое оценивается на заданном временном интервале (15-20 измерений, 15-20 сек.). Из практики применения статистических метрик следует, что отклонение метрики-индикатора более чем в 2–3 раза от среднего значения на интервал прогноза (160 тестовых значений) является признаком аномалии. Для выявления аномалии во временном ряду контролируемой величины (загрузка процессора, объем потребляемой оперативной памяти и др.) при экспериментальной отработке алгоритма можно также использовать стандартное отклонение и среднее абсолютное отклонение (median absolute deviation – MAD).

Несмотря на то, что очевидный тренд в изменении признака (время ответа сервера HTTP) отсутствует, тестовый и обучающие наборы насыщены выбросами, квадратичное отклонение для нормализованной величины колеблется в интервале (0,4) и, судя по графику на рис.3, модель нейросети позволяет однозначно фиксировать аномалию.

На рис.4 и рис.5 представлены исходные данные и результаты повторного эксперимента. В качестве исходных данных для формирования обучающего и тестового наборов используется вновь сгенерированный временной ряд параметра времени ответа сервера HTTP со значительными выбросами и отсутствием тренда (рис. 4). Начиная с 630 секунды в данных обнаружена аномалия.

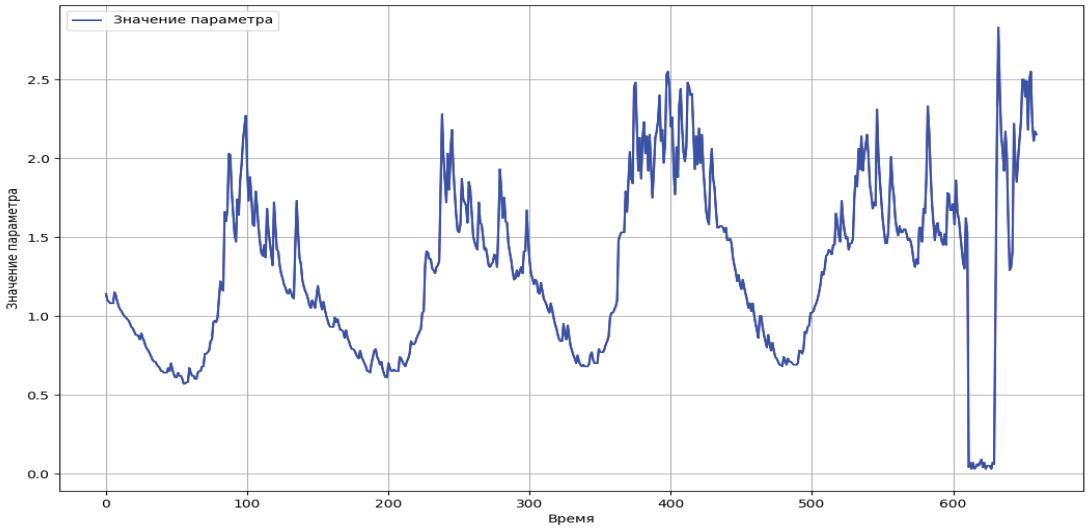

Рис. 6. Исходный набор данных для формирования тестовой и обучающей выборок – степень использования процессора, 660 измерений

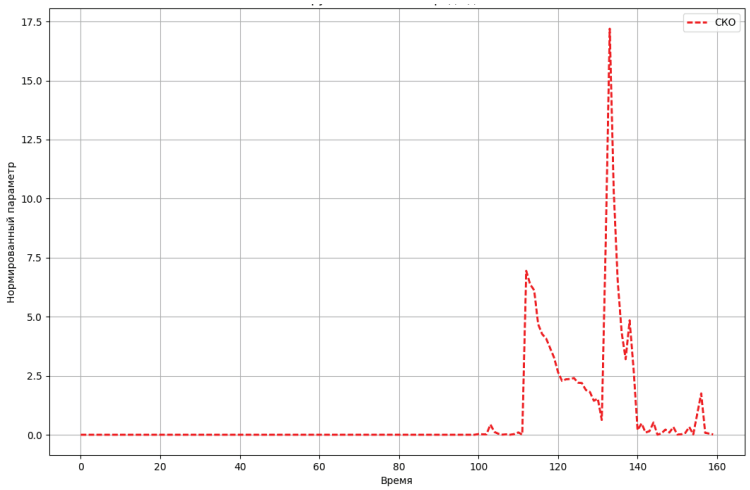

Рис. 7. Результаты моделирования работы алгоритма обнаружения аномалий с использованием FFN нейросети

Рассмотренная выше модель нейросети не является единственной для решения задач обнаружения аномалий и ПТВ. Авторами статьи разработана программа, реализующая алгоритм, в котором в качестве базовой модели нейросети использована сеть прямого распространения (Feed Forward Neural Network, FFN), имеющая следующие характеристики: количество слоев – 3, количество элементов в n-граммах – 100, функция активации нейронов – сигмоида, алгоритм оптимизации – Adam. Программа разработана на языке Go с использованием библиотеки Go-deep в ОС Astra Linux SE 1.6. Проведенные эксперименты показали, что по окончанию процесса обучения нейросети (950 эпох) среднеквадратичная ошибка (MSE), используемая в качестве функции потерь при обучении нейросети, составила 0,0004 – 0,0013, что свидетельствует о высоком уровне настройки нейросети.

На рис. 6 представлен исходный набор данных, содержащий 660 измерений значения степени загрузки процессора, которые получены с использованием среды *osquery* и сохранены в файле в формате CSV. В данные внесена аномалия, начиная с 600-ой секунды. При этом присутствует явный тренд в изменениях значений признака.

Результат моделирования работы алгоритма с указанной выше структурой сети на наборе данных, представленном на рисунке 6, изображен на рис. 7.

Как видно из рис. 7, признаком аномалии (ПТВ) является значение СКО, превышающее 7.5.

Результаты проведенных экспериментов показали, что представленные в статье метод и его алгоритмическая и программная реализации позволяют решить задачу обнаружения аномального поведения контролируемых систем посредством прогнозирования и мониторинга анализируемых параметров, выход значений которых за установленные пределы является признаком ПТВ и СИБ. Важным достоинством данного подхода является наличие возможности адаптации нейросети в случае изменения режима и условий функционирования системы.

Заключение

Представленные в статье модели, алгоритм и программное обеспечение предназначены для автоматизации процессов обнаружения событий информационной безопасности и программно-технических воздействий на КСИИ. Метод и алгоритм реализованы в виде программ на языках Python3 и Go с использованием пакетов Keras, Tensorflow, osquery, go-deep. Результаты экспериментов подтвердили работоспособность данного подхода и целесообразность его применения для обнаружения кибератак в реальном масштабе времени. Реализация данного позволит повысить оперативность обнаружения программно-технических воздействия и достоверность принятия решения о мерах по нейтрализации их последствий.

Направлением дальнейших исследований является полноценная реализация и отладка данного метода на языке Go для ОС Astra Linux и проведение экспериментов для изучения и анализа возможностей, характеристик и способов эффективного применения различных типов нейросетей: LSTM, генеративно-состязательной нейросети (GAN, generative adversarial network), управляемого рекуррентного блока (GRU, gated recurrent unit) для решения задачи обнаружения аномалий в режиме реального времени.

Литература

1. Raghavendra Chalapathy, Sanjay Chawla. Deep learning for anomaly detection: A survey. https://arxiv.org/pdf/1901.03407.pdf. (дата обращения: 25.07.2022).
2. Liu Hua Yeo, Xiangtong Che, Shalini Lakkaraju. Understanding Modern Intrusion Detection Systems: A Survey. – URL: https://arxiv.org/pdf/1708.07174 (дата обращения: 01.09.2022).
3. Созыкин А.В. Обзор методов обучения глубоких нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2017. Т. 6, № 3. С. 28–59. DOI: 10.14529/cmse170303.
4. Гайфулина Д.А., Котенко И.В. Применение методов глубокого обучения в задачах кибербезопасности. Часть 2. Вопросы кибербезопасности. 2020. № 4(38) . DOI:10.21681/2311-3456-2020-04-11-21.
5. Yang Xin, Mingcheng Gao, Haixia Hou. Machine Learning and Deep Learning Methods for Cybersecurity https://www.researchgate.net/publication/325159145. (дата обращения: 25.08.2022).
6. On the Effectiveness of Machine and Deep Learning for Cyber Security Giovanni Apruzzese, Michele Colajanni, Luca Ferretti. 2018 10th International Conference on Cyber Conflict CyCon. https://ccdcoe.org/uploads/2018/10/Art-19-On-the-Effectiveness-of-Machine-and-Deep-Learning-for-Cyber-Security.pdf. (дата обращения: 25.08.2022).
7. Pushpa Iyer, Tanvi Jadhav. Analysis of Modern Intrusion Detection Algorithms and Developing a Smart IDS, 2021 International Conference on Intelligent Technologies (CONIT). – URL: https://ieeexplore.ieee.org/document/ 9498519/ (дата обращения: 30.07.2022).
8. Yin C., Zhu Y., Liu S., Fei J., Zhang H. An Enhancing Framework for Botnet Detection Using Generative Adversarial Networks // 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD). IEEE, 2018. P. 228-234. (дата обращения: 5.09.2022).
9. Зунин, В. В. Intel OpenVINO™ Toolkit: анализ производительности выполнения генеративно-состязательных нейронных сетей / В. В. Зунин, А. Ю. Романов // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). – 2021. – № 2. – С. 83-90. – DOI: 10.31114/2078-7707-2021-2-83-90. – EDN QWXODC.
10. Chen H., Jiang L. GAN-based method for cyber-intrusion detection // arXiv preprint arXiv:1904.02426, 2019. P. 1-6. (дата обращения: 25.08.2022).
11. Yin C., Zhu Y., Fei J., He X. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks // IEEE Access, 2017. Vol. 5. P. 21954-21961. (дата обращения: 5.09.2022).
12. Zhu M., Ye K., Wang Y., Xu C.Z. A Deep Learning Approach for Network Anomaly Detection Based on AMF-LSTM // IFIP International Conference on Network and Parallel Computing Springer, Cham, 2018. P. 137-141. (дата обращения: 5.09.2022).
13. Manavi M., Zhang Y. A New Intrusion Detection System Based on Gated Recurrent Unit (GRU) and Genetic Algorithm // International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage, Springer, Cham, 2019. P. 368-383. (дата обращения: 5.09.2022).
14. Зуев В.Н. Обнаружение аномалий сетевого трафика методом глубокого обучения Программные продукты и системы / Software & Systems 1 (34) 2021. Т. 34. № 1. С. 091–097. DOI: 10.15827/0236-235X.133.091-097
15. Нейросетевая технология обнаружения аномального сетевого трафика / В. А. Частикова, С. А. Жерлицын, Я. И. Воля, В. В. Сотников // Прикаспийский журнал: управление и высокие технологии. – 2020. – № 1(49). – С. 20-32. – DOI 10.21672/2074-1707.2020.49.4.020-032. – EDN WUCDII.
16. Кажемский М.А., Шелухин О.И. Многоклассовая классификация сетевых атак на информационные ресурсы методами машинного обучения // Труды учебных заведений связи. 2019. Т. 5. № 1. С. 107-115.
17. Моделирование идентификации профиля кибератак на основе анализа поведения устройств в сети. Болодурина И.П., Парфёнов Д.И., Забродина Л.С. и др. Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». 2019. Т. 19, № 4. С. 48–59.
18. Нечахин В. А., Пищик Б. Н. Применение методов глубинного обучения для обнаружения вторжений // Вестник НГУ. Серия: Информационные технологии. 2019. Т. 17, №2. С. 114–121. DOI 10.25205/1818-7900-2019-17-2-114-121
19. Jain G., Sharma M., Agarwal B. Optimizing semantic LSTM for spam detection // International Journal of Information Technology. 2019. Vol. 11. No. 2. P. 239-250.
20. Jason Brownlee How to Develop LSTM Models for Time Series Forecasting. – URL: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/ (дата обращения: 10.09.2022).
21. Alex Graves Generating Sequences with Recurrent Neural Networks. University of Toronto (2014). – URL: https://arxiv.org/pdf/1308.0850v5.pdf (дата обращения: 10.09.2022).

DETECTING INFORMATION SECURITY INCIDENTS BASED ON NEURAL NETWORK TECHNOLOGY

Bukin A.V.[7], Samonov A.V.[8], Tihonov E.I.[9]

Objective: *develop model, algorithmic and software for detecting in real time attempts to disrupt the correct functioning of critical information infrastructure systems with neural network technologies.*

Methods *analysis of modern machine learning methods and neural network technologies, synthesis and modeling of correct behavior of programs, algorithmization of learning processes and application of neural networks, experimental studies of developed algorithms and programs on the stand.*

Study results: *The characteristics of machine learning methods and neural network technologies used to detect software and technical impacts and information security incidents are given. The method for solving this problem based on neural networks with LSTM and FFN architectures has been developed. The description of the algorithm and fragments of the software implementation of the method in the programming languages Python3 and Go using Tensorflow and Keras libraries is given. An important advantage of the proposed approach is the possibility of adapting the neural network in the event of a change in the mode and conditions of operation of the system. The results obtained during the experiments indicate the possibility and expediency of using this approach to detect software and technical impacts on critical information infrastructure systems on a time scale close to real with a high level of reliability.*

Scientific novelty: *consists in the application of deep learning technology based on a long-term short-term neural network LSTM, which has the ability to adapt to changing modes and conditions, to solve the problem of detecting signs of a violation of the correct functioning of nodes of information and telecommunications systems in real time.*

Keywords: *anomaly detection, deep learning, intrusion detection systems, loss function, machine learning methods, recurrent neural networks, time series*

References

1. Raghavendra Chalapathy, Sanjay Chawla. Deep learning for anomaly detection: A survey. https://arxiv.org/pdf/1901.03407.pdf. (Data obrashhenija: 25.07.2022).
2. Liu Hua Yeo, Xiangtong Che, Shalini Lakkaraju. Understanding Modern Intrusion Detection Systems: A Survey. – URL: https://arxiv.org/pdf/1708.07174 (data obrashhenija: 01.09.2022).
3. Sozykin A.V. Obzor metodov obuchenija glubokih nejronnyh setej // Vestnik JuUrGU. Serija: Vychislitel'naja matematika i informatika. 2017. T. 6, № 3. S. 28–59. DOI: 10.14529/cmse170303.
4. Gajfulina D.A., Kotenko I.V. Primenenie metodov glubokogo obuchenija v zadachah kiberbezopasnosti. Chast' 2. Voprosy kiberbezopasnosti. 2020. № 4(38) . DOI:10.21681/2311-3456-2020-04-11-21.
5. Yang Xin, Mingcheng Gao, Haixia Hou. Machine Learning and Deep Learning Methods for Cybersecurity https://www.researchgate.net/publication/325159145. (Data obrashhenija: 25.08.2022).
6. On the Effectiveness of Machine and Deep Learning for Cyber Security Giovanni Apruzzese, Michele Colajanni, Luca Ferretti. 2018 10th International Conference on Cyber Conflict CyCon. https://ccdcoe.org/uploads/2018/10/Art-19-On-the-Effectiveness-of-Machine-and-Deep-Learning-for-Cyber-Security.pdf. (Data obrashhenija: 25.08.2022).
7. Pushpa Iyer, Tanvi Jadhav. Analysis of Modern Intrusion Detection Algorithms and Developing a Smart IDS, 2021 International Conference on Intelligent Technologies (CONIT). – URL: https://ieeexplore.ieee.org/document/ 9498519/ (Data obrashhenija: 30.07.2022).
8. Yin C., Zhu Y., Liu S., Fei J., Zhang H. An Enhancing Framework for Botnet Detection Using Generative Adversarial Networks // 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD). IEEE, 2018. P. 228-234. (Data obrashhenija: 5.09.2022).
9. Zunin, V. V. Intel OpenVINO™ Toolkit: analiz proizvoditel'nosti vypolnenija generativno-sostjazatel'nyh nejronnyh setej / V. V. Zunin, A. Ju. Romanov // Problemy razrabotki perspektivnyh mikro- i nanojelektronnyh sistem (MJeS). – 2021. – № 2. – S. 83-90. – DOI: 10.31114/2078-7707-2021-2-83-90. – EDN QWXODC.
10. Chen H., Jiang L. GAN-based method for cyber-intrusion detection // arXiv preprint arXiv:1904.02426, 2019. P. 1-6. (Data obrashhenija: 25.08.2022).

7 Alexander V. Bukin, research scientist, Mozhaiskiy Military Space Academy St.Petersburg, Russia. E-mail: bukinav.mail@gmail.com

8 Alexander V. Samonov, Ph.D. in technical sciences, associate professor, senior research scientist Mozhaiskiy Military Space Academy St.Petersburg, Russia. Email: a.samonov@mail.ru ORCID: 0000-0002-0390-4481

9 Edward I. Tihonov, Ph.D. in technical sciences, associate professor, senior research scientist Mozhaiskiy Military Space Academy. E-mail: inta.et@gmail.com

11. Yin C., Zhu Y., Fei J., He X. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks // IEEE Access, 2017. Vol. 5. P. 21954-21961. (data obrashhenija: 5.09.2022).
12. Zhu M., Ye K., Wang Y., Xu C.Z. A Deep Learning Approach for Network Anomaly Detection Based on AMF-LSTM // IFIP International Conference on Network and Parallel Computing Springer, Cham, 2018. P. 137-141. (Data obrashhenija: 5.09.2022).
13. Manavi M., Zhang Y. A New Intrusion Detection System Based on Gated Recurrent Unit (GRU) and Genetic Algorithm // International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage, Springer, Cham, 2019. P. 368-383. (data obrashhenija: 5.09.2022).
14. Zuev V.N.. Obnaruzhenie anomalij setevogo trafika metodom glubokogo obuchenija Programmnye produkty i sistemy / Software & Systems 1 (34) 2021. T. 34. № 1. S. 091–097. DOI: 10.15827/0236-235X.133.091-097
15. Nejrosetevaja tehnologija obnaruzhenija anomal'nogo setevogo trafika / V. A. Chastikova, S. A. Zherlicyn, Ja. I. Volja, V. V. Sotnikov // Prikaspijskij zhurnal: upravlenie i vysokie tehnologii. – 2020. – № 1(49). – S. 20-32. – DOI 10.21672/2074-1707.2020.49.4.020-032. – EDN WUCDII.
16. Kazhemskij M.A., Sheluhin O.I. Mnogoklassovaja klassifikacija setevyh atak na informacionnye resursy metodami mashinnogo obuchenija // Trudy uchebnyh zavedenij svjazi. 2019. T. 5. № 1. S. 107-115.
17. Modelirovanie identifikacii profilja kiberatak na osnove analiza povedenija ustrojstv v seti. Bolodurina I.P., Parfjonov D.I., Zabrodina L.S. i dr. Vestnik JuUrGU. Serija «Komp'juternye tehnologii, upravlenie, radiojelektronika». 2019. T. 19, № 4. S. 48–59.
18. Nechahin V. A., Pishhik B. N. Primenenie metodov glubinnogo obuchenija dlja obnaruzhenija vtorzhenij // Vestnik NGU. Serija: Informacionnye tehnologii. 2019. T. 17, №2. S. 114–121. DOI: 10.25205/1818-7900-2019-17-2-114-121
19. Jain G., Sharma M., Agarwal B. Optimizing semantic LSTM for spam detection // International Journal of Information Technology. 2019. Vol. 11. No. 2. P. 239-250.
20. Jason Brownlee How to Develop LSTM Models for Time Series Forecasting. – URL: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/ (Data obrashhenija: 10.09.2022).
21. Alex Graves Generating Sequences with Recurrent Neural Networks. University of Toronto (2014). – URL: https://arxiv.org/pdf/1308.0850v5.pdf (Data obrashhenija: 10.09.2022).