УДК 004.056.5 **DOI** 10.15622/ia.21.6.9

И.В. Котенко, И.Б. Саенко, О.С. Лаута, А.М. Крибель

МЕТОДИКА ОБНАРУЖЕНИЯ АНОМАЛИЙ И КИБЕРАТАК НА ОСНОВЕ ИНТЕГРАЦИИ МЕТОДОВ ФРАКТАЛЬНОГО АНАЛИЗА И МАШИННОГО ОБУЧЕНИЯ

Котенко И.В., Саенко И.Б., Лаута О.С., Крибель А.М. **Методика обнаружения аномалий и кибератак на основе интеграции методов фрактального анализа и машинного обучения.**

Аннотация. В современных сетях передачи данных для постоянного мониторинга сетевого трафика и обнаружения в нем аномальной активности, а также идентификации и классификации кибератак, необходимо учитывать большое число факторов и параметров, включая возможные сетевые маршруты, времена задержки данных, потери пакетов и новые свойства трафика, отличающиеся от нормальных. Все это является побудительным мотивом к поиску новых методов и методик обнаружения кибератак и защиты от них сетей передачи данных. В статье рассматривается методика обнаружения аномалий и кибератак, предназначенная для использования в современных сетях передачи данных, которая основывается на интеграции методов фрактального анализа и машинного обучения. Методика ориентирована на выполнение в реальном или близком к реальному масштабе времени и включает несколько этапов: (1) выявления аномалий в сетевом трафике, (2) идентификации в аномалиях кибератак и (3) классификации кибератак. Первый этап реализуется с помощью методов фрактального анализа (оценки самоподобия сетевого трафика), второй и третий – с применением методов машинного обучения, использующих ячейки рекуррентных нейронных сетей с долгой краткосрочной памятью. Рассматриваются вопросы программной реализации предлагаемой методики, включая формирование набора данных, содержащего сетевые пакеты, циркулирующие в сети передачи данных. Представлены результаты экспериментальной оценки предложенной методики, полученные с использованием сформированного набора данных. Результаты экспериментов показали достаточно высокую эффективность предложенной методики и разработанных для нее решений, позволяющих осуществлять раннее обнаружение как известных, так и неизвестных кибератак.

Ключевые слова: кибератака, фрактальный анализ, показатель Херста, машинное обучение, LSTM.

1. Введение. Мировые тенденции в области информатизации и связи на базе цифровых методов передачи, обработки, хранения, представления и защиты информации заключаются во взаимном проникновении и «сращивании» информационных и телекоммуникационных систем не только на уровне технологий их разработки и эксплуатации, но и их структурного и функционального объединения. При этом широко используется термин «сеть передачи данных» (СПД).

Интеграция и конвергенция сетей и служб связи в современных СПД обеспечивает доступ пользователей к любой услуге связи за счет гибких возможностей по их обработке и управлению. По этой

причине, с одной стороны, повышаются эффективность СПД, в том числе устойчивость функционирования СПД, и экономическая выгода от использования СПД. С другой стороны, это предоставляет злоумышленникам возможность воздействовать на СПД путем реализации кибератак (КА).

Воздействие КА возможно за счет массового использования устаревших операционных систем, малоэффективных механизмов защиты и наличия множественных уязвимостей в незащищенных сетевых протоколах. Используя подобные уязвимости, злоумышленники могут изменять настройки сетевых устройств, прослушивать и перенаправлять трафик, блокировать сетевое взаимодействие и получать несанкционированный доступ к внутренним компонентам СПД.

Воздействие КА приводит к появлению в сетевом трафике аномальной активности [1, 2]. Для постоянного мониторинга сетевого трафика и обнаружения в нем аномальной активности, идентификации и классификации атак, а также выявления в нем ложных изменений необходимо учитывать наличие большого количества параметров, характеризующих проявление новых свойств трафика. Однако при этом остается необходимость обеспечения высокого качества обслуживания приложений. Все это является побудительным мотивом для поиска новых методов и методик обнаружения КА и защиты от них СПД. К их числу можно отнести предлагаемую в настоящей статье методику, основанную на интеграции методов фрактального анализа и машинного обучения. Методы фрактального анализа позволяют оперативно выявлять аномальный трафик, а методы машинного обучения обеспечивают идентификацию, классификацию и прогнозирование КА.

Ключевым параметром фрактального анализа является показатель Херста, или показатель масштабирования (scaling). Эту меру, как правило, используют при анализе временных рядов. Чем больше задержка между двумя одинаковыми парами значений во временном ряду, тем меньше показатель Херста. При этом выдвигается гипотеза, что для нахождения показателя Херста достаточно знать, стационарен исследуемый процесс или нет. От этого зависит выбор алгоритма для дальнейшего вычисления данного показателя.

Анализ показал, что одним из достаточно эффективных методов идентификации, классификации и прогнозирования КА является использование искусственных нейронных сетей типа LSTM (Long Short-Term Memory). Свойство рекуррентности, присущее нейронным

сетям LSTM, позволяет им «обращаться» к результатам своей работы в прошлом и делать анализ предсказаний. Тем самым контекст решений по выработке мероприятий по защите от КА в будущем будет зависеть не только от первичного обучения сетей LSTM, но и от их дальнейшей работы в потоке поступающих данных [3].

Таким образом, с целью идентификации и классификации КА сначала следует определить, является трафик стационарным или нестационарным. Далее следует рассчитать показатель Херста и определить наличие в трафике свойства самоподобия. Изменение значения показателя Херста говорит о появлении в сетевом трафике аномалий, вызванных КА. На дальнейших этапах происходит идентификация и классификация КА, а также выработка мероприятий по защите СПД с применением LSTM [4, 5].

В настоящее время вопросы, связанные с изучением самоподобных свойств временных рядов и их практическим применением в различных системах мониторинга, находятся в фокусе внимания многих исследователей. Так, в [6–8] для выявления закономерностей во временных рядах использовался метод R/S-анализа (rescaled range analysis). В [9] моделировался и исследовался на самоподобие трафик VoIP (Voice Over Internet Protocol). В [10–12] изучался не только показатель Херста, но и фрактальная размерность. В [13, 14] было дано объяснение, почему телекоммуникационный трафик обладает фрактальными свойствами.

При этом следует отметить, что существует мало практических экспериментов, направленных на изучение фрактальных свойств трафика. Среди такого рода исследований можно выделить работы [15–17]. Однако в [15] трафик рассматривается не в СПД, а в радиоволнах, передаваемых сотовыми станциями. В [18–20] исследователи пришли к выводу о самоподобии транспортного трафика. При этом они полагались исключительно на визуальные знаки, отыскивая на графиках похожие участки и выдавая их за самоподобные процессы.

Одним из первых исследований, в котором было обращено внимание на свойство самоподобия трафика СПД, является работа [9]. Кроме того, следует указать ряд исследований, в которых аномалии сетевого трафика и КА определяются на основе оценки энтропии [21–24], а также применения методов машинного обучения [25–33].

Так, в [21] предложен алгоритм для обнаружения резких изменений во временных рядах энтропии сети, который непрерывно проводит краткосрочные прогнозы, определяет разницу между прогнозами и фактическим наблюдаемым значением энтропии. Чем

выше разница, тем более резким является изменение. В [22] для обнаружения атак используется подход, основанный на энтропии всех полезных атрибутов сетевых пакетов во время КА. В [23] эмпирически оценивается энтропия Хартли, энтропия Шеннона, энтропия Реньи и обобщенная энтропия. В работе [24] предложен инструмент для внедрения аномалий в заданную трассировку потока. Этот инструмент был апробирован для внедрения аномалий, образованных от следующий трех типов КА: сканировании сети, смещение входа и отказ в обслуживании. Однако указанные работы направлены на обнаружение только тех КА, которые приводят к резкому изменению в сетевом трафике СПД (например, DDoS, черви и сканирование сети).

В [25] рассматривается метод обнаружения аномалий трафика сети и классификации КА, основанный на использовании многослойной нейронной сети состояния эхо-сигнала. При этом используются результаты вычисления статистического распределения и корреляции характеристик сетевого потока. В работе [26] для обнаружения КА сетевой поток размечается и преобразуется в последовательности «слов», которые формируют «предложения», отражающие взаимодействие между компьютерами. Далее с применением рекуррентной нейронной сети с долгой краткосрочной памятью изучается семантическая и синтаксическая грамматика предлагаемого языка для прогнозирования связи между двумя IP-адресами, причем ошибка прогнозирования используется в качестве показателя того, насколько типичны или нетипичны наблюдаемые коммуникации.

Преимущества методов машинного обучения перед другими методами обнаружения сетевых атак были рассмотрены в работе [27] на примере следующих методов: деревья решений; байесовские сети; сплайны; алгоритмы кластеризации и регрессии. Показано, что они обладают возможностями обнаруживать не только аномалии, но и злоупотребления. В [28] предлагается в целях повышения эффективности обнаружения сетевых атак комплексировать нейронные, иммунные и нейро-нечеткие классификаторы.

В [29] показано, что за последние 10 лет машинное обучение стало играть еще большую роль в многочисленных приложениях кибербезопасности. Они защищают киберпространство от атак, позволяют обнаруживать вторжения, спам и вредоносные программы. При этом следует отметить, что методы машинного обучения – это не обязательно нейронные сети. К числу наиболее известных и популярных методов машинного обучения по-прежнему относятся Support Vector Machine (SVM) («машина опорных векторов») [30],

Random Forest (RF) («случайный лес») [31], Decision Tree («дерево решений») [32], k-Nearest Neighbors («k ближайших значений») [33] и многие другие. Некоторые из перечисленных методов будут использованы и исследованы в настоящей работе.

Таким образом, настоящая работа, с одной стороны, опирается на достигнутые успехи в исследовании самоподобных свойств трафика СПД. С другой стороны, она развивает известные решения в направлении создания методики, позволяющей обнаруживать аномалии сетевого трафика, вызванные КА.

Целью настоящей работы является разработка методики обнаружения аномалий и КА, основывающейся на интеграции методов фрактального анализа и машинного обучения и позволяющей за счет этого достигнуть достаточно высоких скорости и точности обнаружения как известных, так и неизвестных КА.

Вклад настоящей работы заключается в следующем: (1) реализован подход к обнаружению КА, основанный на анализе фрактальных свойств трафика; (2) исследованы структуры долговременных зависимостей в трафике СПД, позволяющие выявлять его характерные особенности в интересах раннего обнаружения КА; (3) обоснована структура сети LSTM, позволяющая идентифицировать и классифицировать КА с достаточно высокой вероятностью; (4) разработан программный прототип, реализующий предлагаемую методику, и сгенерирован набор данных с трафиком СПД, содержащим аномалии от воздействия как известных, так и неизвестных КА; (5) проведена экспериментальная оценка предлагаемой методики, показывающая ее достаточно высокую эффективность.

Новизна полученных результатов заключается в том, что на основе экспериментальных исследований обоснован наилучший метод определения самоподобия для нестационарных и стационарных временных рядов, позволяющий с высокой точностью и достаточно быстро обнаруживать изменения в трафике, а также определена структура сети LSTM, позволяющая с высокой точностью и достаточно быстро прогнозировать факт воздействия КА, на основе которого в дальнейшем могут вырабатываться проактивные мероприятия защиты.

2. Основные теоретические положения методики. Собираемый сетевой трафик представляется в виде временного ряда $X = \{x_1, x_2, \dots, x_n\}$, описанного через фрактальное броуновское движение (ФБД). Случайный процесс $X(t)$, соответствующий ФБД, использует параметр Херста H, $0 \le H \le 1$. Приращение этого

процесса $\Delta X(\tau) = X(t+\tau) - X(t)$ имеет следующее нормальное распределение:

$$P(\Delta X(\tau) < x) = \frac{1}{\sqrt{2\pi}\delta_0 \tau^{2H}} \int_{-\infty}^{x} \exp\left[-\frac{z^2}{2\delta_0^2 \tau^{2H}}\right] dz, \quad (1)$$

где δ_0 – коэффициент диффузии.

Показатель H характеризует степень самоподобия процесса. Чем ближе этот параметр к единице, тем более ярко проявляются фрактальные свойства. Равенство $H = 0{,}5$ говорит об отсутствии самоподобия [1, 9, 10]. ФБД с $H = 0.5$ совпадает с классическим броуновским движением, что делает временной ряд наиболее зашумленным.

Обнаружение аномалий в сетевом трафике с помощью фрактального анализа происходит следующим образом. Вначале определяют, является ли трафик стационарным. Для этой цели используется тест Дики-Фуллера [18]. Далее вычисляется значение H одним из методов, в зависимости от того, является трафик стационарным или нет. Если трафик является стационарным, то используется метод R/S (Rescaled Range Analysis) [35]. Если трафик является нестационарным, то используется метод DFA (Detrended Fluctuation Analysis) [36]. Если значение H лежит в диапазоне [0,5; 1], то трафик считается нормальным, т.е. в нем отсутствуют аномалии. В противном случае считается, что трафик является аномальным, т.е. он содержит аномалии.

Кроме методов фрактального анализа существует множество других способов, позволяющих определить аномалии во временном ряду. К числу таких методов можно отнести, например, методы авторегрессионного интегрированного скользящего среднего, кумулятивных сумм, SVM, RF и некоторые другие. Указанные методы неплохо справляются с обнаружением аномальных выбросов. При таких выбросах аномалия проявляется в виде нестационарности некоторых наблюдаемых временных рядов. Эти аномалии проявляются не только в виде мгновенных скачков амплитуды измерений, но и как медленные тренды, практически невидимые за время наблюдений. Однако при тестировании вышеуказанных алгоритмов на реальном трафике СПД оказалось, что не всегда имеющиеся в трафике выбросы являются аномальными. Поэтому в рассматриваемой методике предлагается дополнительно использовать для обнаружения аномальных выбросов метод машинного обучения, основанный на применении гибридной искусственной нейронной сети, состоящей из автокодировщика (autoencoder) и классификатора.

Под автокодировщиком понимается нейронная сеть прямого распространения, которая восстанавливает входной сигнал на выходе. Внутри у него имеется скрытый слой, который представляет собой код, описывающий некоторую модель. Автокодировщик конструируются таким образом, чтобы иметь возможность точно скопировать вход на выходе.

Обучение автокодировщика осуществляется по схеме, представленной на рисунке 1. Алгоритм обучения автокодировщика включает следующие шаги.

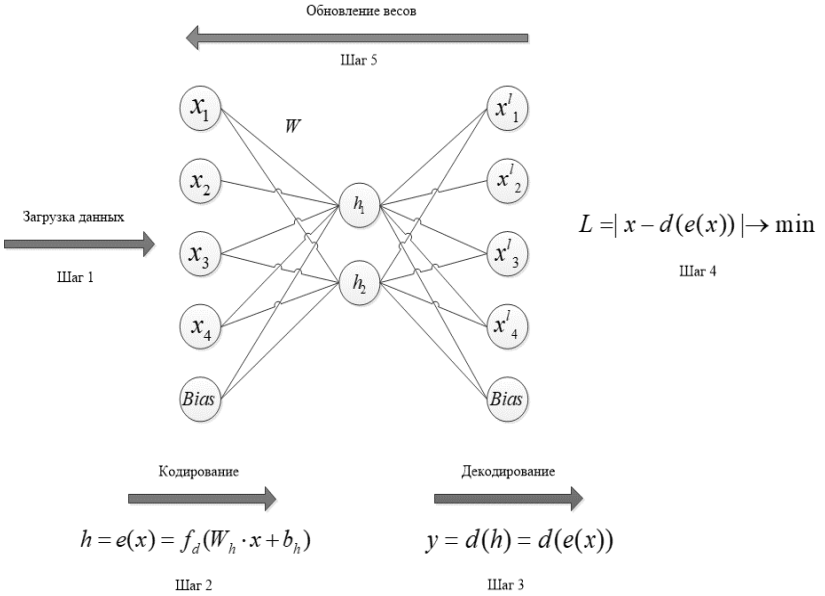


Рис. 1. Схема обучения автокодировщика

Шаг 1 (загрузка данных): входные данные x подаются на вход нейронной сети.

Шаг 2 (кодирование): кодировщик кодирует входной вектор x в вектор h меньшего размера:

$$h = e(x) = f_d(W_h \cdot x + b_h), \quad (2)$$

где f_d – функция активации (ReLU, сигмоидальная функция или гиперболический тангенс) промежуточного слоя; b_h – вектор

смещений промежуточного слоя; W_h – матрица весов промежуточного слоя.

Шаг 3 (декодирование данных): вектор h декодируется, чтобы воссоздать ввод:

$$y = d(h) = f_e(W_y \cdot h + b_y). \quad (3)$$

Параметры f_e, W_y, b_y аналогичны соответствующим параметрам входного слоя. Выход будет иметь такой же размер, что и вход.

Шаг 4 (расчет ошибки): вычисляется ошибка:

$$L = |x - d(e(x))| \rightarrow min. \quad (4)$$

Ошибка L определяет разницу между входным и выходным вектором. Цель процедуры обучения автокодировщика заключается в минимизации этой ошибки.

Шаг 5 (обновление весов): с помощью алгоритма обратного распространения ошибки следует обновить веса $W = \{W_h, W_y\}$.

Шаги 1-5 повторяются до тех пор, пока ошибка не снизится до приемлемого результата.

После обучения автокодировщик может восстанавливать наблюдения с достаточно малой ошибкой. Однако когда он попытается предсказать/реконструировать аномальное наблюдение, он обнаружит, что никогда не видел таких последовательностей во время обучения. Таким образом, ошибка восстановления между исходными и восстановленными данными будет выше для аномальных данных, чем для обычных.

В качестве функции потерь выступает категориальная кросс-энтропия (Categorical Cross-Entropy):

$$CCE(y, p) = -\sum_{i=1}^{M}(y_i \cdot \log_2 p_i), \quad (5)$$

где p_i – прогнозируемая вероятность выходной метки y_i; M – количество классов.

В случае, если $M = 2$, т.е. реализуется бинарная классификация, как в нашем случае, формула (5) преобразуется в формулу, описывающую бинарную кросс-энтропию (Binary Cross-Entropy):

$$BCE(y_i, p_i) = -y_i \cdot \log_2 p_i + (1 - y_i)\log_2(1 - p_i). \quad (6)$$

Функцией, определяющей выходное значение нейрона в зависимости от результата взвешенной суммы входов и порогового значения, является функция мягкого максимума (SoftMax). Она является обобщенной логистической функцией для многомерного случая и вычисляется следующим образом:

$$f(p_i) = \frac{e^{p_i}}{\sum_{i=1}^{M} e^{p_i}}. \quad (7)$$

3. Общая структура и реализация методики обнаружения аномалий и классификации КА в СПД.

Общая структура. Предлагаемая методика обнаружения аномалий и классификации КА в СПД включает пять этапов (рисунок 2):

1) сбор сетевого трафика;

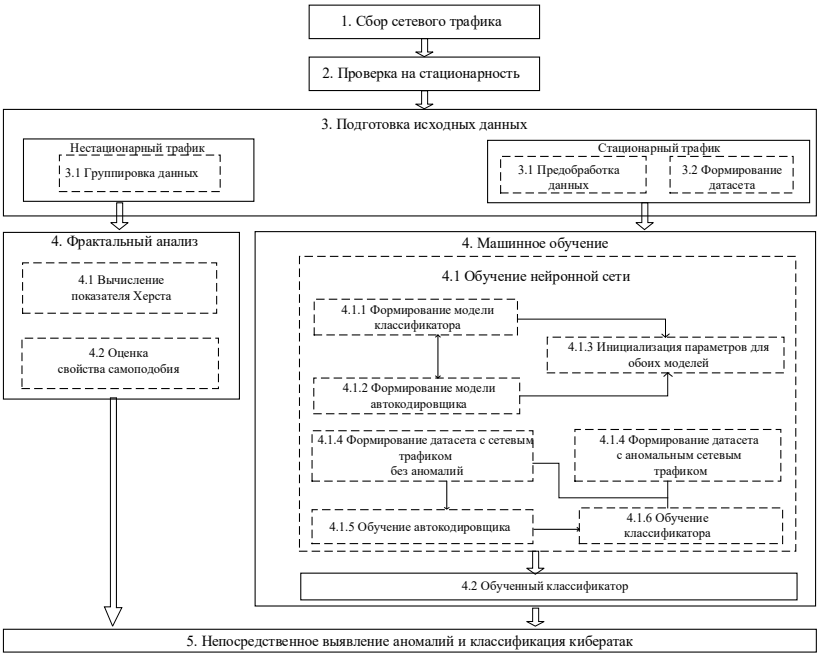


Рис. 2. Общая структура методики выявления аномалий и классификации КА в СПД

2) проверка трафика на стационарность;
3) подготовка исходных наборов данных;
4) фрактальный анализ и машинное обучение;
5) непосредственное выявление аномалий и классификация кибератак с помощью обученного классификатора.

После сбора сетевого трафика осуществляется его проверка на стационарность с использованием теста Дики-Фуллера. Подготовка исходных наборов данных (датасетов) необходима как для стационарного, так и для нестационарного случая. В нестационарном случае исходные наборы данных используются для вычисления показателя Херста методом DFA. Эти оценки используются на следующем этапе для выявления аномалий сетевого трафика путем сравнения текущего и предварительно рассчитанного эталонного показателей Херста.

В стационарном случае исходные наборы данных применяются для обучения автокодировщика, использующего LSTM-ячейки, и связанного с ним классификатора.

На заключительном этапе осуществляется целевая обработка трафика, заключающаяся в непосредственном выявлении аномалий по результатам фрактального анализа (для стационарного и нестационарного трафика) и классификации атак с помощью обученных автокодировщика и классификатора (для стационарного трафика).

Реализация. Для реализации разработанной методики обнаружения аномалий и классификации КА в СПД разработан программный прототип, схема работы которого представлена на рисунке 3.

Прототип разработан на языке Python. Выбор этого языка был обусловлен тем, что он ориентирован на повышение производительности разрабатываемого кода, поддерживает многопоточные вычисления и имеет большое количество библиотек. В частности, использовались библиотека универсального назначения Pandas и библиотека NumPy, позволяющий работать с многомерными массивами (тензорами) и математическими функциями. Для построения графиков применялся модуль Matplotlib. Необходимые расчеты проводились в интегрированной среде разработки Jupiter notebook.

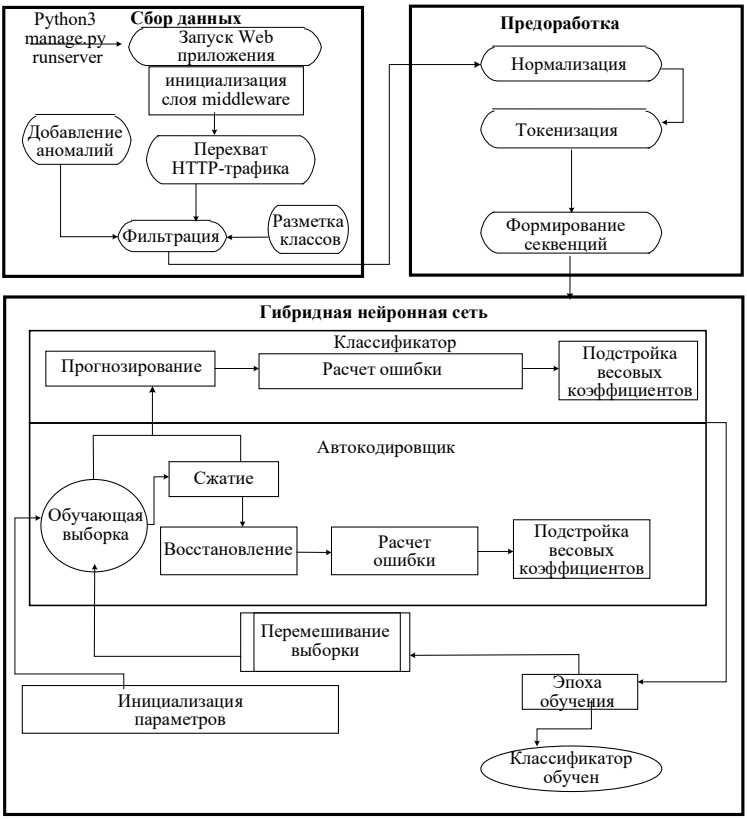


Рис. 3. Схема работы программного прототипа, реализующего методику

Программный прототип включает в себя 3 подсистемы:

1) сбора данных;
2) предобработки сформированного набора данных;
3) гибридная нейронная сеть.

Главной задачей первой подсистемы является сбор, обработка и анализ сформированного набора данных. Сбор данных, предназначенных для обучения нейронной сети, осуществлялся следующим образом. На языке Python было написано Web-приложение, способное перехватывать любые пользовательские запросы с помощью промежуточного программного слоя (middleware). Такой подход позволяет обрабатывать запросы из браузера прежде, чем они достигнут представления сервера Django, а также ответы от представлений до того, как они возвращаются в браузер.

Перехваченные запросы записываются в лог-файл (журнализируются). Для старта Web-приложения запускается HTTP-сервер с интерфейсом шлюза веб-сервера Python. Для этой цели вводится команда `python manage.py runserver`.

В набор данных, сформированный после сбора HTTP трафика, добавляются аномальные запросы для задания мультиклассовости КА. К такого рода запросам относятся: SQL Injection, Cross-Site Scripting, Cross-Site Request Forgery, XML External Entity Injection, CRLF Injection и HTTP Response Splitting. Каждый тип аномального запроса помечается в наборе данных как отдельный класс.

Главной задачей второй подсистемы является нормализация полученного набора данных. Для этого запросы оборачиваются специальными токенами <START> и <STOP>. Это позволяет задавать верное вероятностное распределение над последовательностями разной длины. С помощью теста Дики-Фуллера производится оценка стационарности получившегося ряда путем нахождения распределения длин между двумя одинаковыми символами.

После нормализации данные токенизируются. Поскольку HTTP является текстовым протоколом, для токенизации используется векторное представление символов. Для этой цели сперва осуществляется замена символов, встречающихся в наборе данных, на числовой эквивалент, который не имеет самостоятельного значения для внешнего или внутреннего использования. Затем слова переводятся в последовательность секвенций, т.е. пронумерованный набор объектов, среди которых допускаются повторения, причем порядок объектов имеет значение. Нумерация происходит натуральными числами

При этом учитывается то, что все секвенции должны быть одной длины. Если запрос меньше длины секвенции, то оставшиеся символы заполняются нулями.

Главной задачей третьей подсистемы является обучение нейронной сети и выявление аномалий в СПД. Для этого первоначально инициализируются гиперпараметры гибридной нейронной сети, и происходит ее обучение на сформированных секвенциях. Классификатор гибридной нейронной сети ищет закономерности в данных, привязывая их к размеченным классам на этапе обучения. Данные, которые классификатор не смог отнести к какому-либо классу, в том числе и к классу легитимных данных, помечаются как аномальные. Они соответствуют атакам «нулевого дня».

Анализ аномалий в стационарной сети. Для выявления аномалий в стационарной сети предлагается использовать гибридную нейронную сеть, модель которой показана на рисунке 4.

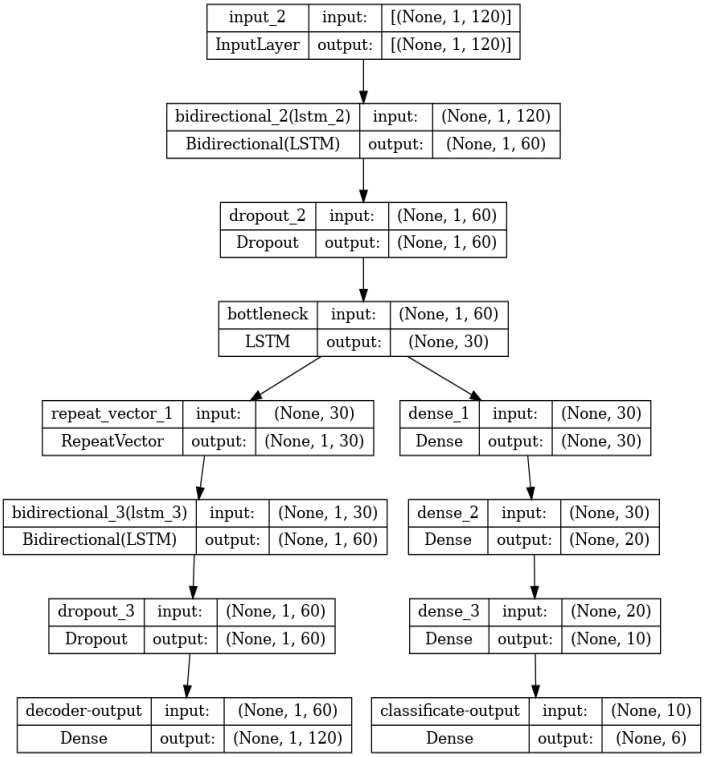


Рис. 4. Модель гибридной нейронной сети

Гибридная нейронная сеть имеет две ветви и соответствующие им два выхода.

Гибридная сеть имеет различные по своему назначению слои. Наиболее характерными слоями являются следующие:

– *Dropout* (отсеивать) – этот слой предназначен для решения проблемы переобучения в нейронных сетях;

– *Bidirectional* (двунаправленный) – это слой позволяет осуществлять генеративное глубокое обучение, при котором выходной слой может одновременно получать информацию из прошлого (назад) и будущего (вперед) состояний;

– *Bottleneck* (узкое место) – этот слой позволяет уменьшать количество свойств и, соответственно, количество операций в каждом слое, что обеспечивает высокую скорость получения результата.

Входной слой гибридной нейронной сети имеет 120 нейронов, применяющихся как для автокодировщика, так и для классификатора.

В качестве слоев автокодировщика используются ячейки LSTM [34]. Сети LSTM являются подтипом более общих рекуррентных нейронных сетей. Ключевым атрибутом таких нейронных сетей является их способность сохранять информацию (состояние ячейки) для ее дальнейшего использования. LSTM может удалять информацию из состояния ячейки. Этот процесс регулируется фильтрами. Они позволяют пропускать информацию на основании некоторых условий. Фильтры состоят из слоя сигмоидальной нейронной сети и операции поточечного умножения. Сигмоидальный слой возвращает числа от нуля до единицы, определяющие, какую долю каждого блока информации следует пропускать дальше по сети. Ноль в данном случае означает «не пропускать ничего», единица – «пропустить все».

Свойство рекуррентности позволяет искусственной нейронной сети «обращаться» к результатам своей работы в прошлом, делать анализ предсказаний. Тем самым контекст решений в будущем будет зависеть не только от первичного глубокого обучения LSTM, но и ее дальнейшей работы в потоке [37, 38].

В процессе обучения на входной слой гибридной нейронной сети поступают различные вектора (рисунок 5) (падасеквенции, секвенции или эмбеддинги) в зависимости от реализации. Различными оттенками серого цвета отражаются разные типы атак – SQL инъекции, несанкционированный доступ, несанкционированный доступ с кодированием, XXE-инъекция и внедрение шаблонов на стороне сервера. Падасеквенция – это функция библиотеки Tensorflow, которая используется для того, чтобы все последовательности в списке имели одинаковую длину. По умолчанию это делается путем добавления нуля в начало каждой последовательности, пока каждая последовательность не будет иметь ту же длину, что и самая длинная последовательность. А для преобразования положительных целых чисел (индексов) в плотные векторы фиксированного размера применяется первый слой нейронной сети – эмбеддинг (Embedding). Эмбеддинги требуют больших вычислительных ресурсов. Поэтому для быстрых вычислений, как показали исследования, лучше применять падасеквенции длиной 120 символов.

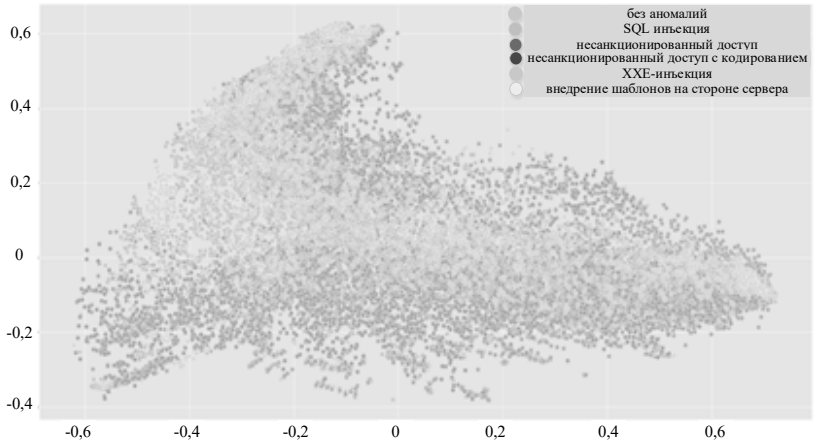


Рис. 5. Визуальное представление векторного представления данных, подаваемых на вход гибридной нейронной сети

В середине гибридной нейронной сети число нейронов уменьшается до 30. Это приводит к потере информации, так как из 120 нейронов не вся информация попадает на 30 нейронов.

У классификатора (правая ветвь) на последнем слое имеется шесть нейронов, которые соответствуют шести размеченным классам, упомянутым выше. Если классификатор не может отнести данные ни к одному из классов с вероятностью больше 0,6, то такой запрос отмечается подозрительным (может считаться атакой нулевого дня). У автокодировщика на последнем слое имеется 120 нейронов. Он приводит информацию, содержащуюся на 30 нейронах, к первоначальному виду. В результате в середине слоя сохраняется только самая важная информация, из которой можно восстановить информацию в исходном виде – «скрытые латентные представления».

Пример отображения скрытых латентных представлений показан на рисунке 6. Такие представления позволяют классификатору находить дополнительные закономерности в данных. Это существенно уменьшает ложные срабатывания, что, в свою очередь, повышает вероятность обнаружения атак нулевого дня.

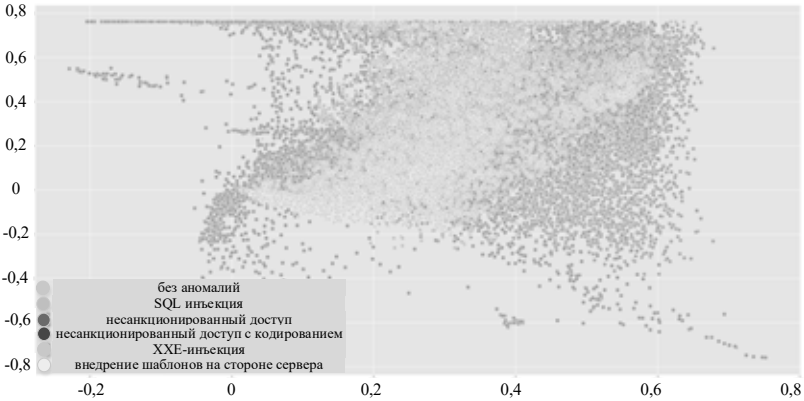


Рис. 6. Скрытые латентные представления, полученные в результате сжатия информации автокодировщиком

4. Экспериментальная оценка методики выявления аномалий и КА в СПД. Для экспериментальной оценки рассматриваемой методики разработан киберполигон, представленный на рисунке 7.

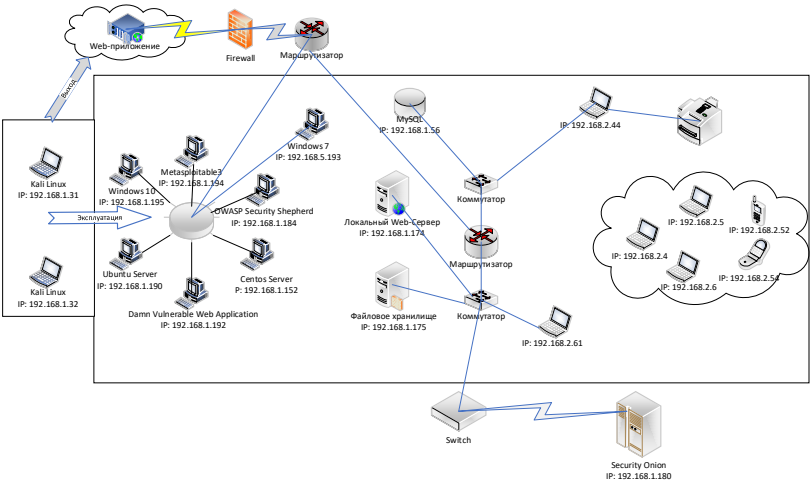


Рис. 7. Киберполигон для сбора и анализа защищенности сетевого трафика

На киберполигоне проведено около 30 видов кибератак и сгенерировано 40 Гб легитимного трафика. Сетевой трафик записывался в pcap-файлы. Из этого трафика формировался набор

данных с помощью Netsniff-ng и Bro. Атаки проводились с помощью дистрибутива Kali Linux на заведомо уязвимые сервисы, развернутые в центральной части схемы. Далее проводился поиск аномальных всплесков с помощью методов кумулятивных сумм, RF и SVM.

Несмотря на то, что эти алгоритмы прекрасно справляются с задачами поиска аномальных всплесков, было обнаружено, что не всегда всплески являются аномалиями. Для передачи пакетов данных рассматривались современные стандарты, протоколы и технологии построения высокоскоростных СПД. Сценарий, по которому происходила передача пакетов сообщений, в этом случае являлся стационарным. При этом предполагалось, что СПД обладает свойством самоподобия. Это предположение в дальнейшем было подтверждено в ходе экспериментов. Также полагалось, что порты в оборудовании пограничной сети имеют пропускную способность 1 Гбит/с и работают по протоколу Ethernet [1].

Генерация трафика осуществлялась с помощью имитационной модели, разработанной в среде симулятора GNS3.

Перечень основных атрибутов, которые были включены в сгенерированный набор данных, представлены в таблице 1. Общее количество различных значений параметра Flow.ID в наборе данных оказалось равным 1522917. Анализ самоподобия проводился по временному ряду, образованному из значений поля Packet.Length.Mean. Этот атрибут в сгенерированном наборе данных имел 10700 уникальных значений. Наиболее часто встречались значения 267,5 и 243,5 [1].

На смоделированную сетевую инфраструктуру воздействовали два типа КА. Этими атаками были DDoS-атака и «Сканирование сети и ее уязвимостей» [39, 40]. Трафик под воздействием атаки первого типа моделировался с помощью тестового оборудование для IP-сетей IXIA. Для реализации кибератак первого типа использовались распределенная сеть и методы SYN Flood, Ping Flood и UDP Flood. Второй тип атаки моделировался с помощью средств сканирования IP-сетей Nmap и Xspider. Для реализации этой атаки использовался метод зондирования.

Анализ сетевого трафика в условиях воздействия указанных КА показал, что многие всплески являются легитимными. С другой стороны, во многих местах, где отсутствуют всплески, имеются аномалии. Поэтому необходимо эффективное обнаружение всплесков трафика, выделение из них аномальных, классификация выявленных аномалий с целью прогнозирования факта воздействия КА и выработки мероприятий по противодействию КА.

Таблица 1. Атрибуты сгенерированного набора данных

№	Имя атрибута	Комментарии
Атрибуты общего назначения		
1	Timestamp	Момент захвата пакета
2	Protocol	Идентификатор протокола транспортного уровня
3	Flow.ID	Идентификатор потока
4	Flow.Duration	Общая продолжительность потока
Атрибуты, характеризующие длину пакета в прямом направлении		
5	Fwd.Packet.Length.Max	Максимальная длина пакета в прямом направлении
6	Fwd.Packet.Length.Mean	Средняя длина пакета в прямом направлении
7	Fwd.Packet.Length.Min	Минимальная длина пакета в прямом направлении
8	Fwd.Packet.Length.SD	Стандартное отклонение длины пакета в прямом направлении
Атрибуты, характеризующие длину пакета в обратном направлении		
9	Bwd.Packet.Length.Max	Максимальная длина пакета в обратном направлении
10	Bwd.Packet.Length.Mean	Средняя длина пакета в обратном направлении
11	Bwd.Packet.Length.Min	Минимальная длина пакета в обратном направлении
12	Bwd.Packet.Length.SD	Стандартное отклонение длины пакета в обратном направлении
Атрибуты источника и получателя пакетов		
13	Source.IP	IP-адрес источника потока
14	Source.Port	Номер порта источника
15	Destination.IP	IP-адрес получателя
16	Destination.Port	Номер порта получателя
Атрибуты с обобщенными данными		
17	Packet.Length.Mean	Среднее значение длины пакетов, зарегистрированных в потоке в прямом и обратном направлениях
18	Total.Fwd.Packets	Общее количество прямых пакетов
19	Total.Backward.Packets	Общее количество обратных пакетов
20	Total.Length.of.Fwd	Общее количество байтов в прямом направлении, полученных от всего потока
21	Total.Length.of.Backward	Общее количество байтов в обратном направлении, полученное из всего потока

Сетевой трафик, полученный с применением кибернетического полигона, разделялся на легитимные и аномальные выборки. Для

каждой выборки вычислялся показатель Херста по алгоритмам R/S и DFA.

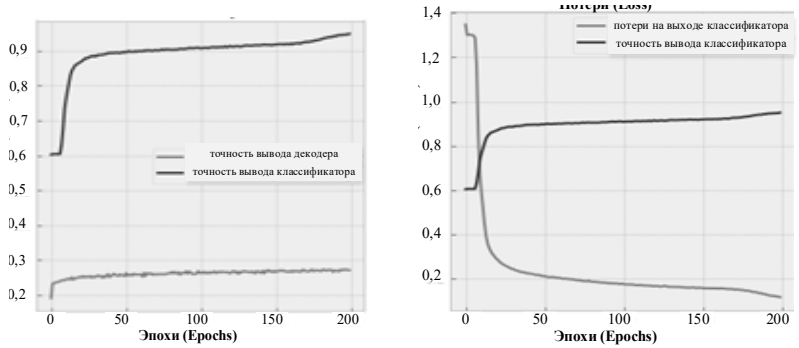
Пример вычисления H для нестационарного легитимного и аномального трафика UDP подробно описан в [1, 10].

Обучающий набор данных для идентификации и классификации атак включал как легитимный, так и аномальный трафик.

На вход автокодировщика подавался только легитимный трафик. На вход классификатора поступал легитимный и аномальный трафик, а также скрытые латентные представления, полученные от автокодировщика после кодирования информации. Подбор параметров осуществлялся таким образом, чтобы функция потерь при обучении автокодировщика уменьшалась, в то время как точность классификатора росла.

Для оценки точности и полноты обнаружения аномалий на обученной нейронной сети вначале использовался набор данных с кибератаками, применявшийся для обучения сети. Точность обнаружения аномалий в этом случае составила 96,9 %.

На рисунке 8 показано, как изменялись точность классификатора и функция потерь за 200 циклов (эпох) обучения. Точность классификатора (рисунок 8а) стремительно возрастает приблизительно на 15-ой эпохе обучения. На 200-ой эпохе она приближается к единице. В то же время функция потерь (рисунок 8б) уменьшается (особенно сильно – с 15-ой эпохи) и на 200-ой эпохе стремится к нулю.



а) б)

Рис. 8. Зависимости точности (Accuracy) и потерь (Loss) от эпох обучения при обучении декодера и классификатора

Затем был сформирован новый набор данных, который включал атаки, ранее неизвестные классификатору (атаки нулевого дня). Было распознано 80% неизвестных ранее атак нулевого дня, и было верно определено, что 99% легитимных запросов не являются аномальными. Кроме того, было замечено, что возможны ложные срабатывания. В частности, было отброшено всего 2 запроса. Учитывая тот факт, что в наборе данных содержится 57000 запросов, из которых 20000 являются аномальными, можно сделать вывод, что данный факт не относится к существенным недостаткам рассматриваемой методики.

Помимо эксперимента по обнаружению аномалий был проведен эксперимент по классификации КА. Одним из методов оценки качества работы классификатора и выбора дискриминационного порога для разделения классов является ROC-кривая (кривая ошибок) (рисунок 9).

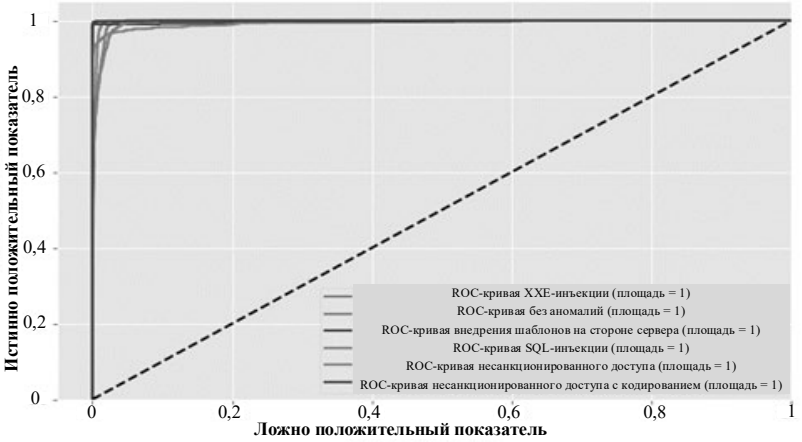


Рис. 9. ROC-кривая для мультиклассовой классификации

ROC-кривая описывает взаимосвязь между чувствительностью модели машинного обучения (true positives rate – доля истинно положительных примеров) и ее специфичностью (true negative rate – доля ложно положительных результатов).

Преимуществом ROC-кривой является ее инвариантность относительно отношения ошибки первого и второго рода. Оптимальное местонахождение точек ROC-кривой – в левом верхнем углу графика, где специфичность и чувствительность находятся на оптимальных уровнях. Именно такое положение ROC-кривой и было зафиксировано в проводимых экспериментах.

Площадь под ROC-кривой определяет точность классификации модели машинного обучения. Чем больше эта площадь, тем больше расхождение между истинными и ложными срабатываниями и тем выше эффективность процесса классификации. Иными словами, чем ближе площадь под ROC-кривой к единице, тем лучше. Практически для всех видов КА, учитываемый в экспериментах, площадь под ROC-кривой была приближенно равна единице.

Визуальный анализ подтвердил высокую точность предложенного подхода с минимальным числом ложных срабатываний. Для более качественной оценки были выделены основные показатели классификации атак (рисунок 10): точность (precision), полнота (recall) и F-мера (f1-score). F-мера является средним гармоническим между точностью и полнотой и играет роль комплексного показателя, позволяющего оценить эффективность классификации атак.

	precision	recall	f1-score	support
xxe	0.83	0.95	0.88	1292
no anomaly	0.99	1.00	0.99	9266
server side template injection	1.00	0.99	0.99	1265
sql injection	0.96	0.93	0.95	1271
traversal	0.94	0.80	0.86	1263
encoding traversal	0.99	0.99	0.99	1305
accuracy			0.97	15662
macro avg	0.95	0.94	0.94	15662
weighted avg	0.97	0.97	0.97	15662

Рис. 10. Основные показатели классификации атак

Из рисунка 10 видно, что для различных видов атак эффективность их классификации находилась в диапазоне от 0,88 (для XXE-инъекции) до 0,99 (для атак внедрения шаблона на стороне сервера и несанкционированного доступа с кодированием).

При этом следует заметить, что разработанная методика не только имеет высокую эффективность классификации КА, но также имеет высокую эффективность обнаружения легитимных запросов (обозначено на рисунке как «no anomaly»), равную 0,99. Этот факт сам по себе говорит о минимизации ложных срабатываний (ошибок первого рода), связанных с отнесением легитимных запросов к атакам, и наоборот.

Сравнительная оценка предложенной методики с другими аналогичными методами и методиками показала, что в ней

обеспечивается достаточно высокая скорость обнаружения известных КА (рисунок 11).

Так, предложенной методике, как и сигнатурному методу, для обнаружения известных КА с вероятностью, превышающей 0,96, достаточно 5 секунд работы на компьютере стандартной конфигурации. В то же время статистическим методам и методам машинного обучения для этого требуется порядка 30 секунд. Это связано с тем, что статистические методы используют накопленную статистику, а эффективность методов машинного обучения зависит от используемых ими моделей классификации и кластеризации [41, 42]. В методах машинного обучения на этапе обучения используется обучающая выборка (train-выборка), а эффективность обнаружения КА оценивается и проверяется на основе тестовой выборки (test-выборки).

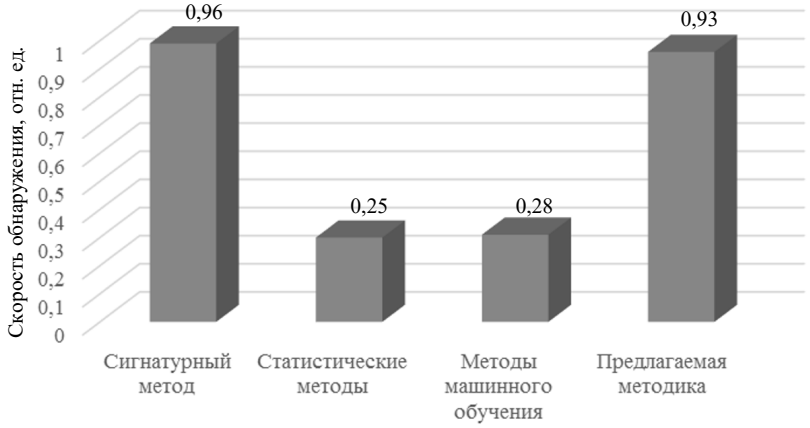


Рис. 11. Сравнительная оценка скорости обнаружения известных КА

Если проводить сравнительную оценку методов и методик по эффективности обнаружения неизвестных КА, то можно утверждать, что в этом случае разработанная методика не уступает по своей эффективности методам машинного обучения, которые в настоящее время демонстрируют в этой области наилучшие результаты по сравнению с сигнатурным и статистическими методами (рисунок 12). Так, за время своей работы, равное 5 секунд, предлагаемая методика и сигнатурный метод позволяют обнаружить неизвестные КА с вероятностями 0,8 и 0,5 соответственно.

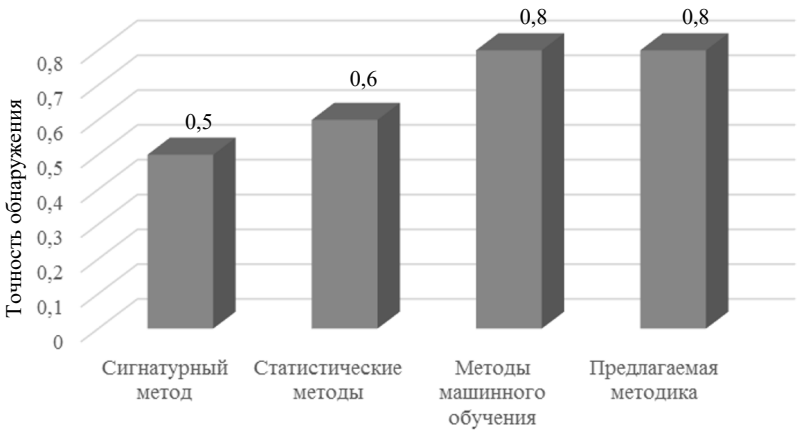


Рис. 12. Сравнительная оценка точности обнаружения неизвестных КА

Статистические методы и методы машинного обучения требуют на решение этой задачи, как было указано выше, порядка 30 секунд. Однако точность обнаружения неизвестных атак с помощью статистических методов менее 0,5, а у методов машинного обучения она может превышать 0,8.

Следует отметить, что дополнительным преимуществом предложенной методики является реализованная в ней возможность обнаружения аномалий в трафике любого рода. Остальные известные методы хорошо работают только в случае стационарного движения.

К числу других достоинств этого подхода следует отнести нетребовательность к системным ресурсам. Кроме того, предложенный подход универсален за счет представления процессов в виде временных рядов. Тип протокола передачи информации, а также вид передаваемой информации (служебная информация, синхронизация, полезная информация) никак не влияют на время определения коэффициента Херста. Он инвариантен к типам деструктивных воздействий и не требует настройки или адаптации к обнаружению конкретных видов атак, в том числе ранее неизвестных.

При этом следует отметить, что увеличение количества обрабатываемых параметров заголовка протокола передачи данных (длина пакета, флаги и т.д.) приводит к увеличению времени вычислений.

5. Заключение. В статье предложена обладающая высокой оперативностью и точностью методика обнаружения аномалий и КА, в которой выявление аномалий в сетевом трафике производится путем

оценки его свойства самоподобия в реальном масштабе времени или близком к реальному, а обнаружение КА и их классификация осуществляются с применением гибридной нейронной сети, в основе построения которой базируются ячейки LSTM. Специфика предлагаемой методики заключается в том, что обнаружение КА выполняется с использованием автокодировщика, обученного на основе эталонных данных работы сети и обмена информацией в ней с учетом всех отклонений от штатного режима работы сети.

В проведенных экспериментах рассматриваемая методика продемонстрировала довольно высокую точность раннего обнаружения КА, достигнув значения 0,93 для известных атак и 0,8 для заранее неизвестных атак.

Дальнейшие исследования связаны с интеграцией программной системы, реализующей предлагаемую методику, с другими известными программными средствами защиты, а также способами обнаружения атак и их классификации, имеющимися в арсенале систем компьютерной безопасности.

Литература

1. Kotenko I., Saenko I., Lauta O., Kribel A. An approach to detecting cyber attacks against smart power grids based on the analysis of network traffic self-similarity // Energies. 2020. vol. 13. no. 19. pp. 5031.
2. Al-Jarrah M., Khalaf G., Amin S. PIN Authentication Using Multi-Model Anomaly Detection in Keystroke Dynamics // Proceedings of the 2019 2nd International Conference on Signal Processing and Information Security (ICSPIS). 2019. pp. 1–4.
3. Ageev S., Kotenko I., Saenko I., Kopchak Y. Abnormal Traffic Detection in Networks of the Internet of Things Based on Fuzzy Logical Inference // Proceedings of the IEEE International Conference on Soft Computing and Measurements (SCM). 2015. pp. 5–8.
4. Котенко Д.И., Котенко И.В., Саенко И.Б. Методы и средства моделирования атак в больших компьютерных сетях: состояние проблемы // Труды СПИИРАН. 2012. № 3 (22). С. 5–30.
5. Brezigar-Masten A., Masten I. CART-based selection of bankruptcy predictors for the logit model // Expert Systems with Applications. 2012. vol. 39. no. 11. pp. 10153–10159.
6. Ju X., Chen V.C.P.; Rosenberger J.M., Liu F. Fast knot optimization for multivariate adaptive regression splines using hill climbing methods // Expert Systems with Applications. 2021. no. 171. p. 114565.
7. Ju X., Rosenberger J.M., Chen V.C.P., Liu F. Global optimization on non-convex two-way interaction truncated linear multivariate adaptive regression splines using mixed integer quadratic programming // Information Sciences. 2022. no. 597. pp. 38–52.
8. Ju X., Liu F., Wang Li., Lee W.-J. Wind farm layout optimization based on support vector regression guided genetic algorithm with consideration of participation among landowners // Energy Conversion and Management. 2019. no. 196. pp. 1267–1281.

9. Dang T.D., Sonkoly B., Molnar S. Fractal analysis and modeling of VoIP traffic // Proceedings of the 11th International Telecommunications Network Strategy and Planning Symposium (NETWORKS 2004). 2004. pp. 123–130.
10. Leland W.E., Taqqu M.S., Willinger W., Wilson D.V. On the self-similar nature of Ethernet traffic // SIGCOMM Comput. Commun. 1993. vol. 23. no. 4. pp. 183–193.
11. Raimundo M.S., Okamoto Jr. J. Application of Hurst Exponent (H) and the R/S Analysis in the Classification of FOREX Securities // International Journal of Modeling and Optimization. 2018. no. 8. pp. 116–124.
12. Sánchez-Granero M.J., Fernández-Martínez M., Trinidad-Segovia J.E. Introducing fractal dimension algorithms to calculate the Hurst exponent of financial time series // Eur. Phys. J. B. 2012. vol. 85. no. 86.
13. Kotenko I., Saenko I., Lauta O., Karpov M. Methodology for management of the protection system of smart power supply networks in the context of cyberattacks // Energies. 2021. vol. 14. no. 18. p. 5963.
14. Kotenko I., Saenko I., Lauta O., Kribel A. Ensuring the survivability of embedded computer networks based on early detection of cyber attacks by integrating fractal analysis and statistical methods // Microprocessors and Microsystems. 2022. no. 90. p. 104459.
15. Strelkovskaya I., Solovskaya I., Makoganiuk A. Spline-Extrapolation Method in Traffic Forecasting in 5G Networks // Journal of Telecommunications and Information Technology. 2019. no. 3. pp. 8–16.
16. Carvalho P., Abdalla H., Soares A., Solis P., Tarchetti P. Analysis of the influence of self-similar traffic in the performance of real time applications. URL: citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.599.4041&rep=rep1&type=pdf (дата доступа: 15.07.2022).
17. Fractal Objects and Self-Similar Processes. URL: archive.physionet.org/tutorials/fmnc/node3.html (дата доступа: 15.07.2022).
18. Ruoyu Y., Wang Y. Hurst Parameter for Security Evaluation of LAN Traffic // Information Technology Journal. 2012. no. 11. pp. 269–275.
19. Singh Gulshan M.B., Sharma B., Grover M., Gupta P. TSA: Self-Train Self-Test Algorithm // Proceedings of the 2020 IEEE International Conference for Innovation in Technology (INOCON). 2020. pp. 1–5.
20. Yu Z., Jiang Z., Tan L., Liu H., Yang Q. Rescaled Range Analysis of Vessel Traffic Flow in the Yangtze River // Proceedings of the 2019 5th International Conference on Transportation Information and Safety (ICTIS). 2019. pp. 1–4.
21. Winter P., Lampesberger H., Zeilinger M., Hermann E. On Detecting Abrupt Changes in Network Entropy Time Series // Communications and Multimedia Security. CMS 2011. Lecture Notes in Computer Science. 2011. vol. 7025. pp. 194–205.
22. Sharma S., Sahu S.K., Jena S.K. On Selection of Attributes for Entropy Based Detection of DDoS // Proceedings of the 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2015. pp. 1096–1100.
23. Bhuyan M.H., Bhattacharyya D.K., Kalita J.K. Information metrics for low-rate DDoS attack detection: A comparative evaluation // Proceedings of the 2014 Seventh International Conference on Contemporary Computing (IC3). 2014. pp. 80-84.
24. Brauckhoff D., Wagner A., May M. FLAME: A Flow-Level Anomaly Modeling Engine // Proceedings of the Workshop on Cyber Security and Test. 2008. pp. 1–6.
25. Zhang S.T., Lin X.B., Wu L., Song Y.Q., Liao N.D., Liang Z.H. Network Traffic Anomaly Detection Based on ML-ESN for Power Metering System // Mathematical Problems in Engineering. 2020. vol. 2020. article ID 7219659.
26. Radford B.J., Apolonio L.M., Trias A.J., Simpson J.A. Network Traffic Anomaly Detection Using Recurrent Neural Networks. URL: doi.org/10.48550/arXiv.1803.10769 (дата доступа: 15.07.2022).

27. Браницкий А.А., Котенко И.В. Анализ и классификация методов обнаружения сетевых атак // Труды СПИИРАН. 2016. № 2 (45). С. 207–244.
28. Браницкий А.А., Котенко И.В. Обнаружение сетевых атак на основе комплексирования нейронных, иммунных и нейро-нечетких классификаторов // Информационно-управляющие системы. 2015. № 4 (77). С. 69-77.
29. Shaukat K., Luo S., Varadharajan V., Hameed I.A., Xu M. A Survey on Machine Learning Techniques for Cyber Security in the Last Decade // IEEE Access. 2020. vol. 8. pp. 222310–222354.
30. Chen W.-H., Hsu S.-H., Shen H.-P. Application of SVM and ANN for intrusion detection // Computers & Operations Research. 2005. vol. 32. no. 10. pp. 2617–2634.
31. Hasan M.A.M., Nasser M., Ahmad S., Molla K.I. Feature selection for intrusion detection using random forest // Journal of information security. 2016. vol. 7. no. 03. p. 129.
32. Zhang Y., Wang S., Wu L. Spam detection via feature selection and decision tree // Advanced Science Letters. 2012. vol. 5. no. 2. pp. 726–730.
33. Su M.-Y. Real-time anomaly detection systems for Denial-of-Service attacks by weighted k-nearest-neighbor classifiers // Expert Systems with Applications. 2011. vol. 38. no. 4. pp. 3492–3498.
34. Gers F., Schraudolph N., Schmidhuber J. Learning precise timing with LSTM recurrent networks // Journal of Machine Learning Research. 2002. vol. 3, pp. 115–143.
35. Shaukat S., Ali A., Batool A., Alqahtan, F., Khan J.S., Ahmad A.J. Intrusion Detection and Attack Classification Leveraging Machine Learning Technique // Proceedings of the 2020 14th International Conference on Innovations in Information Technology (IIT). 2020. pp. 198–202.
36. Nurul A.H., Zaheera Z.A., Puvanasvaran A.P., Zakaria N.A., Ahmad R. Risk assessment method for insider threats in cyber security: A review // International Journal of Advanced Computer Science and Applications (ijacsa). 2018. vol. 9. no. 11. pp.16–19.
37. Zhe W.; Wei C., Chunlin L. DoS attack detection model of smart grid based on machine learning method // Proceedings of the 2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS). 2020. pp. 735–738.
38. Karataş G., Akbulut A. Survey on Access Control Mechanisms in Cloud Computing // Journal of Cyber Security and Mobility. 2018. vol. 7. no. 3. pp. 1–36.
39. Lopez J., Rubio J. Access control for cyber-physical systems interconnected to the cloud // Comput. Netw. 2018. vol. 134. no. C. pp. 46–54.
40. Clincy V., Shahriar H. Web Application Firewall: Network Security Models and Configuration // Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). 2018, pp. 835–836.
41. Visoottiviseth V., Sakarin P., Thongwilai J. Choobanjong T. Signature-based and behavior-based attack detection with machine learning for home IoT devices // Proceedings of the 2020 IEEE Region 10 conference (TEN-CON). 2020. pp. 829-834.
42. Amma N.G.B., Selvakumar S., Velusamy R.L. A Statistical Approach for Detection of Denial of Service Attacks in Computer Networks // IEEE Transactions on Network and Service Management. 2020. vol. 17. no. 4. pp. 2511–2522.

Котенко Игорь Витальевич — д-р техн. наук, профессор, заведующий лабораторией, лаборатория проблем компьютерной безопасности, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН); Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики (Университет ИТМО). Область научных интересов: безопасность компьютерных сетей, управление политиками безопасности,

разграничение доступа, аутентификация, анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, анализ и верификация протоколов безопасности и систем защиты информации, защита программного обеспечения от взлома и управление цифровыми правами, технологии моделирования и визуализации для противодействия кибертерроризму. Число научных публикаций — 1000. ivkote@comsec.spb.ru; 14-я линия В.О., 39, 199178, Санкт-Петербург, Россия; р.т.: +7(812)328-7181.

Саенко Игорь Борисович — д-р техн. наук, профессор, ведущий научный сотрудник, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН). Область научных интересов: автоматизированные информационные системы, информационная безопасность, обработка и передача данных по каналам связи, теория моделирования и математическая статистика, теория информации. Число научных публикаций — 400. ibsaen@comsec.spb.ru; 14-я линия В.О., 39, 199178, Санкт-Петербург, Россия; р.т.: +7(812)328–7181.

Лаута Олег Сергеевич — д-р техн. наук, профессор кафедры, кафедра комплексного обеспечения информационной безопасности, Государственный университет морского и речного флота имени адмирала С.О. Макарова. Область научных интересов: защита от компьютерных атак. Число научных публикаций — 184. laos-82@yandex.ru; улица Двинская, 5/7, 198035, Санкт-Петербург, Россия; р.т.: +7(911)842-0228.

Крибель Александр Михайлович — научный сотрудник, лаборатория проблем компьютерной безопасности, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН). Область научных интересов: защита от компьютерных атак. Число научных публикаций — 30. nemo4ka74@gmail.com; 14-я линия В.О., 39, 199178, Санкт-Петербург, Россия; р.т.: +7(812)328-7181.

Поддержка исследований. Работа выполнена при частичной финансовой поддержке бюджетной темы FFZF-2022-0007.

DOI 10.15622/ia.21.6.9

I. KOTENKO, I. SAENKO, O. LAUTA, A. KRIEBEL

ANOMALY AND CYBER ATTACK DETECTION TECHNIQUE BASED ON THE INTEGRATION OF FRACTAL ANALYSIS AND MACHINE LEARNING METHODS

Kotenko I., Saenko I., Lauta O., Kriebel A. **Anomaly and Cyber Attack Detection Technique Based on the Integration of Fractal Analysis and Machine Learning Methods.**

Abstract. In modern data transmission networks, in order to constantly monitor network traffic and detect abnormal activity in it, as well as identify and classify cyber attacks, it is necessary to take into account a large number of factors and parameters, including possible network routes, data delay times, packet losses and new traffic properties that differ from normal. All this is an incentive to search for new methods and techniques for detecting cyber attacks and protecting data networks from them. The article discusses a technique for detecting anomalies and cyberattacks, designed for use in modern data networks, which is based on the integration of fractal analysis and machine learning methods. The technique is focused on real-time or near-real-time execution and includes several steps: (1) detecting anomalies in network traffic, (2) identifying cyber attacks in anomalies, and (3) classifying cyber attacks. The first stage is implemented using fractal analysis methods (evaluating the self-similarity of network traffic), the second and third stages are implemented using machine learning methods that use cells of recurrent neural networks with a long short-term memory. The issues of software implementation of the proposed technique are considered, including the formation of a data set containing network packets circulating in the data transmission network. The results of an experimental evaluation of the proposed technique, obtained using the generated data set, are presented. The results of the experiments showed a rather high efficiency of the proposed technique and the solutions developed for it, which allow early detection of both known and unknown cyber attacks.

Keywords: cyber attack, fractal analysis, Hurst exponent, machine learning, LSTM.

References

1. Kotenko I., Saenko I., Lauta O., Kribel A. An approach to detecting cyber attacks against smart power grids based on the analysis of network traffic self-similarity. Energies. 2020. vol. 13. no. 19. p. 5031.
2. Al-Jarrah M., Khalaf G., Amin S. PIN Authentication Using Multi-Model Anomaly Detection in Keystroke Dynamics. Proceedings of the 2019 2nd International Conference on Signal Processing and Information Security. 2019. pp. 1-4.
3. Ageev S., Kotenko I., Saenko I., Kopchak Y. Abnormal Traffic Detection in Networks of the Internet of Things Based on Fuzzy Logical Inference. Proceedings of the IEEE International Conference on Soft Computing and Measurements (SCM). 2015. pp. 5–8.
4. Kotenko D.I., Kotenko I.V., Saenko I.B. Methods and tools for modeling attacks in large computer networks: the state of the problem. Trudy SPIIRAN – SPIIRAS Proceedings. 2012. no. 3 (22). pp. 5–30. (In Russ.).
5. Brezigar-Masten A., Masten I. CART-based selection of bankruptcy predictors for the logit model. Expert Systems with Applications. 2012. vol. 39. no. 11. pp. 10153-10159.
6. Ju X., Chen V.C.P.; Rosenberger J.M., Liu F. Fast knot optimization for multivariate adaptive regression splines using hill climbing methods. Expert Systems with Applications. 2021. no. 171. p. 114565.

7. Ju X., Rosenberger J.M., Chen V.C.P., Liu F. Global optimization on non-convex two-way interaction truncated linear multivariate adaptive regression splines using mixed integer quadratic programming. Information Sciences. 2022. no. 597. pp. 38-52.
8. Ju X., Liu F., Wang Li., Lee W.-J. Wind farm layout optimization based on support vector regression guided genetic algorithm with consideration of participation among landowners. Energy Conversion and Management. 2019. no. 196. pp. 1267-1281.
9. Dang T.D., Sonkoly B., Molnar S. Fractal analysis and modeling of VoIP traffic. Proceedings of the 11th International Telecommunications Network Strategy and Planning Symposium (NETWORKS 2004). 2004. pp. 123–130.
10. Leland W.E., Taqqu M.S., Willinger W., Wilson D.V. On the self-similar nature of Ethernet traffic. SIGCOMM Comput. Commun. 1993. vol. 23. no. 4. pp. 183-193.
11. Raimundo M.S., Okamoto Jr. J. Application of Hurst Exponent (H) and the R/S Analysis in the Classification of FOREX Securities. International Journal of Modeling and Optimization. 2018. no. 8. pp. 116-124.
12. Sánchez-Granero M.J., Fernández-Martínez M., Trinidad-Segovia J.E. Introducing fractal dimension algorithms to calculate the Hurst exponent of financial time series. Eur. Phys. J. B. 2012. vol. 85. no. 86.
13. Kotenko I., Saenko I., Lauta O., Karpov M. Methodology for management of the protection system of smart power supply networks in the context of cyberattacks. Energies. 2021. vol. 14. no. 18. p. 5963.
14. Kotenko I., Saenko I., Lauta O., Kribel A. Ensuring the survivability of embedded computer networks based on early detection of cyber attacks by integrating fractal analysis and statistical methods. Microprocessors and Microsystems. 2022. no. 90. p. 104459.
15. Strelkovskaya I., Solovskaya I., Makoganiuk A. Spline-Extrapolation Method in Traffic Forecasting in 5G Networks. Journal of Telecommunications and Information Technology. 2019. no. 3. pp. 8-16.
16. Carvalho P., Abdalla H., Soares A., Solis P., Tarchetti P. Analysis of the influence of self-similar traffic in the performance of real time applications. Available at: citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.599.4041&rep=rep1&type=pdf (accessed: 15.07.2022).
17. Fractal Objects and Self-Similar Processes. Available at: archive.physionet.org/tutorials/fmnc/node3.html (accessed: 15.07.2022).
18. Ruoyu Y., Wang Y. Hurst Parameter for Security Evaluation of LAN Traffic. Information Technology Journal. 2012. no. 11. pp. 269–275.
19. Singh Gulshan M.B., Sharma B., Grover M., Gupta P. TSA: Self-Train Self-Test Algorithm. Proceedings of the 2020 IEEE International Conference for Innovation in Technology (INOCON). 2020. pp. 1–5.
20. Yu Z., Jiang Z., Tan L., Liu H., Yang Q. Rescaled Range Analysis of Vessel Traffic Flow in the Yangtze River. Proceedings of the 2019 5th International Conference on Transportation Information and Safety (ICTIS). 2019. pp. 1–4.
21. Winter P., Lampesberger H., Zeilinger M., Hermann E. On Detecting Abrupt Changes in Network Entropy Time Series. Communications and Multimedia Security. CMS 2011. Lecture Notes in Computer Science. vol 7025. 2011. pp. 194–205.
22. Sharma S., Sahu S.K., Jena S.K. On Selection of Attributes for Entropy Based Detection of DDoS. Proceedings of the 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2015. pp. 1096–1100.
23. Bhuyan M.H., Bhattacharyya D.K., Kalita J.K. Information metrics for low-rate DDoS attack detection: A comparative evaluation. Proceedings of the 2014 Seventh International Conference on Contemporary Computing (IC3). 2014. pp. 80–84.

24. Brauckhoff D., Wagner A., May M. FLAME: A Flow-Level Anomaly Modeling Engineю Proceedings of the Workshop on Cyber Security and Test. 2008. pp. 1–6.
25. Zhang S.T., Lin X.B., Wu L., Song Y.Q., Liao N.D., Liang Z.H. Network Traffic Anomaly Detection Based on ML-ESN for Power Metering System. Mathematical Problems in Engineering. 2020. vol. 2020. article ID 7219659.
26. Radford B.J., Apolonio L.M., Trias A.J., Simpson J.A. Network Traffic Anomaly Detection Using Recurrent Neural Networks. Available at: doi.org/10.48550/arXiv.1803.10769 (accessed: 15.07.2022).
27. Branitskiy A.A., Kotenko I.V. Analysis and classification of methods for network attack detection. Trudy SPIIRAN – SPIIRAS Proceedings. 2016. no. 2 (45). pp. 207–244. (In Russ.).
28. Branitskiy A.A., Kotenko I.V. Network Attack Detection Based on Combination of Neural, Immune and Neuro-fuzzy Classifiers. Informacionno-upravlyayushchie sistemy – Information and Control Systems. 2015. no. 4, pp. 152–159. (In Russ.).
29. Shaukat K., Luo S., Varadharajan V., Hameed I.A., Xu M. A Survey on Machine Learning Techniques for Cyber Security in the Last Decade. IEEE Access. 2020. vol. 8. pp. 222310-222354.
30. Chen W.-H., Hsu S.-H., Shen H.-P. Application of SVM and ANN for intrusion detection. Computers & Operations Research. 2005. vol. 32. no. 10. pp. 2617-2634.
31. Hasan M.A.M., Nasser M., Ahmad S., Molla K.I. Feature selection for intrusion detection using random forest. Journal of information security. 2016. vol. 7. no. 03. p. 129.
32. Zhang Y., Wang S., Wu L. Spam detection via feature selection and decision tree. Advanced Science Letters. 2012. vol. 5. no. 2. pp. 726-730.
33. Su M.-Y. Real-time anomaly detection systems for Denial-of-Service attacks by weighted k-nearest-neighbor classifiers. Expert Systems with Applications. 2011. vol. 38. no. 4. pp. 3492-3498.
34. Gers F., Schraudolph N., Schmidhuber J. Learning precise timing with LSTM recurrent networks. Journal of Machine Learning Research. 2002. vol. 3. pp. 115-143.
35. Shaukat S., Ali A., Batool A., Alqahtan, F., Khan J.S., Ahmad A.J. Intrusion Detection and Attack Classification Leveraging Machine Learning Technique. Proceedings of the 2020 14th International Conference on Innovations in Information Technology (IIT). 2020. pp. 198–202.
36. Nurul A.H., Zaheera Z.A., Puvanasvaran A.P., Zakaria N.A., Ahmad R. Risk assessment method for insider threats in cyber security: A review. International Journal of Advanced Computer Science and Applications (ijacsa). 2018. vol. 9. no. 11. pp. 16-19.
37. Zhe W.; Wei C., Chunlin L. DoS attack detection model of smart grid based on machine learning method. Proceedings of the 2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS). 2020. pp. 735–738.
38. Karataş G., Akbulut A. Survey on Access Control Mechanisms in Cloud Computing. Journal of Cyber Security and Mobility. 2018. vol. 7. no. 3. pp. 1-36.
39. Lopez J., Rubio J. Access control for cyber-physical systems interconnected to the cloud. Comput. Netw. 2018. vol. 134. no. C. pp. 46-54.
40. Clincy V., Shahriar H. Web Application Firewall: Network Security Models and Configuration. Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). 2018. pp. 835–836.
41. Visoottiviseth V., Sakarin P., Thongwilai J. Choobanjong T. Signature-based and behavior-based attack detection with machine learning for home IoT devices. Proceedings of the 2020 IEEE Region 10 conference (TEN-CON). 2020. pp. 829-834.

42. Amma N.G.B., Selvakumar S., Velusamy R.L. A Statistical Approach for Detection of Denial of Service Attacks in Computer Networks. 2020. vol. 17. no. 4. pp. 2511-2522.

Kotenko Igor — Ph.D., Dr.Sci., Professor, Head of laboratory, Laboratory of computer security problems, St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS); Saint Petersburg National Research University of Information Technologies, Mechanics and Optics (ITMO University). Research interests: computer network security, security policy management, access control, authentication, network security analysis, intrusion detection, firewalls, deception systems, malware protection, verification of security systems, digital right management, modeling, simulation and visualization technologies for counteraction to cyber terrorism. The number of publications — 1000. ivkote@comsec.spb.ru; 39, 14-th Line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)328-7181.

Saenko Igor — Ph.D., Dr.Sci., Professor, Leading researcher, St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS). Research interests: automated information systems, information security, processing and data transfer on data links, theory of modeling and mathematical statistics, information theory. The number of publications — 400. ibsaen@comsec.spb.ru; 39, 14-th Line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)328–7181.

Lauta Oleg — Ph.D., Dr.Sci., Professor of the department, Department of integrated information security, State University of the Sea and River Fleet named after Admiral S.O. Makarov. Research interests: protection from computer attacks. The number of publications — 184. laos-82@yandex.ru; 5/7, Dvinskaya St., 198035, St. Petersburg, Russia; office phone: +7(911)842-0228.

Kriebel Alexander — Researcher, laboratory of computer security problems, St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS). Research interests: protection against computer attacks. The number of publications — 30. nemo4ka74@gmail.com; 39, 14-th Line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)328-7181.

Acknowledgements. The reported study was partially funded by the budget project FFZF-2022-0007.