УДК 004.07
doi:10.31799/1684-8853-2021-6-42-52

Научные статьи
Articles

Обнаружение атак в сетях интернета вещей методами машинного обучения

Т. М. Татарникова[a], *доктор техн. наук, доцент, orcid.org/0000-0002-6419-0072, tm-tatarn@yandex.ru*
П. Ю. Богданов[a], *старший преподаватель, orcid.org/0000-0002-7533-7316*
[a]*Санкт-Петербургский государственный университет аэрокосмического приборостроения, Б. Морская ул., 67, Санкт-Петербург, 190000, РФ*

Введение: *рост объема цифровых данных, генерируемых в том числе умными устройствами интернета вещей, сделал актуальными исследования, связанные с применением методов машинного обучения для обнаружения аномалий сетевого трафика — наличия сетевых атак.* ***Цель исследования:*** *предложить единый подход к обнаружению атак на разных уровнях архитектуры сети интернета вещей, основанный на методах машинного обучения.* ***Результаты:*** *показано, что на уровне беспроводной сенсорной сети обнаружение атаки связано с выявлением аномального поведения устройства интернета вещей, при котором отклонение поведения устройства интернета вещей от его профиля может расцениваться как компрометация устройства. Построение профилей умных устройств интернета вещей осуществляется на основе статистических характеристик, таких как интенсивность и продолжительность передачи пакетов, доля ретранслированных пакетов и др. На уровне локальной или глобальной проводной сети интернета вещей происходит агрегирование данных, анализ которых также выполняется методами машинного обучения. Обученные классификаторы могут стать частью системы обнаружения сетевых атак, принимающих решение о компрометации узла «на лету». Экспериментальным путем выбраны модели классификаторов сетевых атак как на уровне беспроводной сенсорной сети, так и на уровне локальной или глобальной проводной сети. Лучшие результаты в смысле оценок полноты и точности продемонстрированы методом случайного леса для проводной локальной и (или) глобальной сети и всеми рассмотренными методами для беспроводной сенсорной сети.* ***Практическая значимость:*** *предложенные модели классификаторов могут найти применение при проектировании систем обнаружения атак в сетях интернета вещей.*

Ключевые слова *— сетевая атака, интернет вещей, обучающая выборка, система обнаружения атак, эффективность модели классификатора.*

Для цитирования: Татарникова Т. М., Богданов П. Ю. Обнаружение атак в сетях интернета вещей методами машинного обучения. *Информационно-управляющие системы*, 2021, № 6, с. 42–52. doi:10.31799/1684-8853-2021-6-42-52
For citation: Tatarnikova T. M., Bogdanov P. Yu. Intrusion detection in internet of things networks based on machine learning methods. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 6, pp. 42–52 (In Russian). doi:10.31799/1684-8853-2021-6-42-52

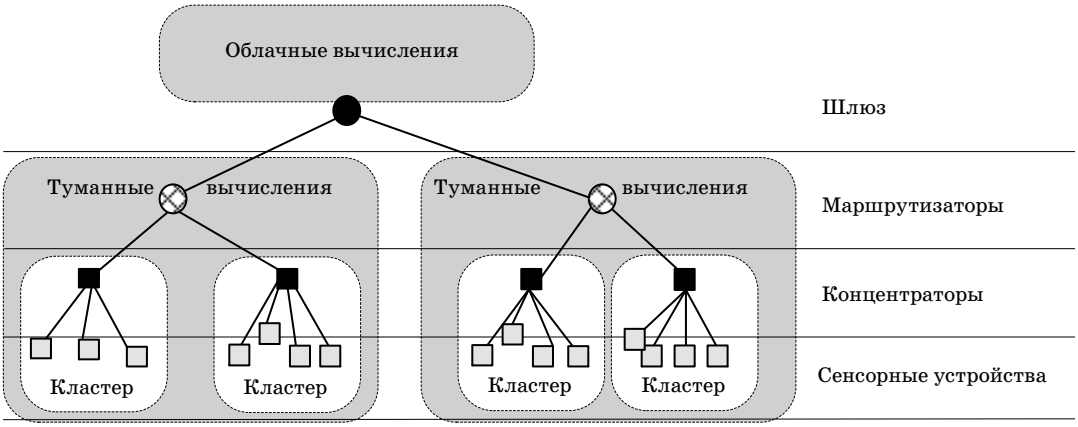
Введение

Сети интернета вещей (Internet of Things — IoT) гетерогенны, мобильны, характеризуются сложной динамической структурой [1]. Эти особенности способствуют развитию сетевых атак, направленных на маршрутизацию, таких как «червоточина», «сборный пункт», «выброс пакетов», Сивиллы, зацикливания, Rush. Последствия от действий атак многоаспектны — от компрометации узлов и захвата контроля над ними до расточения энергии узлов, что приводит к быстрой деградации сети IoT [2, 3].

Защита от атак в сетях IoT реализуется системой обнаружения атак (СОА), которая при своей работе незначительно увеличивает нагрузку на маломощные узлы сети — сенсорные устройства (СУ) [4, 5].

Система обнаружения атак сети IoT имеет иерархическую структуру, как и сама сеть IoT, — три компонентных уровня (рис. 1) [6].

На уровне сенсорных устройств решается задача обнаружения аномального поведения, например резкого увеличения интенсивности и (или) продолжительности передачи, необоснованного снижения уровня остаточной энергии и т. п. Аномальное поведение может свидетельствовать о наличии атаки на IoT-устройство. Общепринятого подхода к обнаружению атак на уровне сенсорных устройств пока не существует — выбор решения зависит от многих причин: на каких устройствах построена сенсорная сеть, являются ли они энергозависимыми, каким атакам подвержены, как долго распространяется атака и пр. Тем не менее анализ источников показал частные решения, основанные на профилировании IoT-устройств. Так, в работе [7] продемонстрировано построение профилей умных устройств интернета вещей на основе статистических характеристик, таких как интенсивность и продолжительность передачи пакетов, доля ретранслированных пакетов. При этом надо учитывать, что профиль IoT-устройства одного разработчика не соответствует профилю IoT-устройства с такими же умными функциями, но другого разработчика. Собственно, в [7] это обстоятельство



■ ***Рис. 1.*** Иерархическая структура организации сетей IoT
■ ***Fig. 1.*** Hierarchical structure of the organization of IoT networks

косвенно демонстрируется — авторы опубликовали в открытом доступе обучающие выборки для нескольких умных дверных звонков, нескольких умных камер (https://archive.ics.uci.edu/ml/machine-learning-databases/00442/). В работе [8] создание профиля IoT-устройства основано на выявлении плотности функции распределения объема переданных и принятых данных.

На уровне сетей — локальных (туманные вычисления) и глобальных (облачные вычисления) — происходит агрегирование трафика, сгенерированного несколькими сотнями или даже миллионами сенсорных устройств, что в общем представляет собой обычный сетевой трафик. Поэтому на уровне сетей задача обнаружения сетевых атак может решаться известными методами, основанными на поведенческой модели сети. Принцип работы таких методов основан на обнаружении несоответствия между текущим режимом работы сети и штатным. Любое несоответствие рассматривается как атака [9, 10]. С другой стороны, поведенческая модель требует времени для фиксации атаки — узловые агенты сначала собирают (накапливают) статистики поведения и передают их в модуль принятия решения СОА. Поскольку распределенные атаки являются атаками реального времени, то для их распознавания и дальнейшего развития необходимы методы, работающие «на лету». В связи с этим при построении СОА в сетях интернета вещей популярность приобретают методы машинного обучения.

В целом объем и разнородность данных, генерируемых интернетом вещей, в настоящее время относят к BigData. Анализ BigData преимущественно выполняется методами машинного обучения [11, 12]. При наличии обученных моделей классификации на входе сетевого узла решение о нормальности/аномальности трафика может приниматься «на лету», в отличие, например, от статистических методов.

Таким образом, объектом исследования является сеть интернета вещей, построенная по иерархическому принципу — от беспроводной сенсорной сети к глобальному облаку. Цель исследования заключается в предложении единого подхода к обнаружению атак на всех уровнях сети IoT, основанного на методах машинного обучения.

Применяемые методы обучения

Для построения модели детектирования угроз на уровне сенсорной сети и в агрегированном трафике применялись следующие методы машинного обучения:

— дерево решений;
— случайный лес;
— нейронная сеть прямого распространения;
— k-ближайших соседей.

Основные параметры обученных классификаторов приведены в табл. 1.

Оценки эффективности моделей классификации

Решение о пригодности или выборе того или иного метода классификации принимается на основе оценки эффективности. При описании оценок используется матрица ошибок (рис. 2).

True positive (истинно положительное решение): результат классификации аномального трафика, предсказанный моделью, совпал с реальной меткой класса.

False Positive (ложноположительное решение): ошибка 1-го рода, модель ошибочно классифицировала нормальный объект, как аномальный.

False Negative (ложноотрицательное решение): ошибка 2-го рода, модель ошибочно классифицировала аномальный объект, как нормальный.

■ ***Таблица 1***. Основные параметры методов машинного обучения
■ ***Table 1***. Basic parameters of machine learning methods

Метод	Основные параметры
Дерево решений [13]	Отсечение ветвей происходит по показателю достоверности — отношению числа неправильно распознанных примеров в листе к общему числу примеров: $C_k = \frac{N_{nk}}{N_k}$, где N_{nk} — число нераспознанных примеров k-го касса; N_k — общее число примеров k-го класса Расщепление происходит по величине наименьшей информационной энтропии $H(x) = -\sum_{i=1}^{N} p(i)\log_2 p(i)$, где x — случайное событие с N возможными состояниями; p — вероятность, что i-й признак станет очередным узлом дерева решений Глубина дерева равна 21
Случайный лес [14]	Ансамбль состоит из 10 деревьев Глубина итогового классификатора 15 Использованы следующие техники построения случайного леса: — bagging — случайная выборка обучающих примеров по равномерному закону с возвратом примеров в исходное обучающее множество, что позволяет избежать переобучения — полного запоминания всех обучающих примеров; — boosting — обучение слабых деревьев решений для сборки их в сильный классификатор, при котором неверно классифицированные примеры обучения получают больший вес, а правильно классифицированные примеры теряют вес, что позволяет при дальнейшем обучении сфокусироваться на ошибочно классифицированных примерах
Нейронная сеть [15, 16]	Один скрытый слой. Количество скрытых слоев выбрано путем каскадного обучения при достижении минимального значения MSE Функция активации нейрона: $\phi(u_k + b_k) = \frac{1}{1+\exp(-\alpha(u_k + b_k))}$, где α — параметр наклона сигмоиды; u_k — взвешенная сумма; b_k — порог Веса меняются согласно локальному градиенту функции ошибки
k-ближайших соседей [4]	*Число ближайших соседей* $k = 3$ Мера близости — евклидова метрика Нормализация параметров методом минимакса: $x_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}$

Класс по алгоритму	Действительный класс	
	+	–
+	True Positive (TP)	False Positive (FP)
–	False Negative (FN)	True Negative (TN)

■ ***Рис. 2***. Матрица ошибок [17]
■ ***Fig. 2***. Matrix of errors [17]

True Negative (истинно отрицательное решение): модель классифицировала объект как нормальный, каким он является в действительности.

Ошибки в чистом виде не используются, поскольку показатели эффективности (качества) алгоритмов являются вероятностными — зависят от обрабатываемых событий и условий [18]. Поэтому для оценки эффективности алгоритмов обучения используются другие критерии, приведенные ниже [19].

Достоверность алгоритма классификации — accuracy:

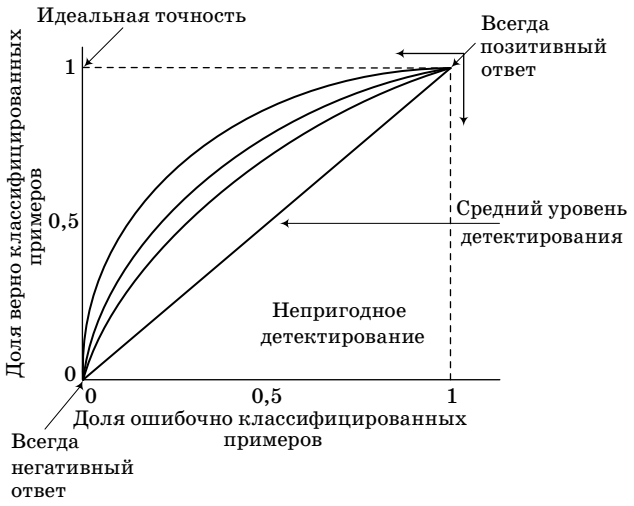
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}.$$

Достоверность показывает в процентах долю верной классификации, в нашем случае трафика, что может характеризовать качество системы обнаружения атак. При такой постановке accuracy 80 % означает, что из 100 записей сетевого трафика верно детектируются 80.

Точность (precision) показывает долю объектов, действительно принадлежащих данному классу относительно всех объектов, которые алгоритм отнес к этому классу:

$$Precision = \frac{TP}{TP + FP}.$$

Доля истинно положительных решений (TPR — True Positive Rate), называемая также



■ ***Рис. 3.*** Примеры ROC-кривых
■ ***Fig. 3.*** Examples of ROC curves

полнотой (recall), определяется как пропорция аномальных экземпляров, классифицированных корректно, во всем множестве аномальных экземпляров, представленных в выборке:

$$TPR = Recall = \frac{TP}{TP + FN}.$$

При оценке эффективности методов глубокого обучения для задач классификации также применяется построение ROC-кривых (Receiver Operating Characteristic Curves). Абстрактно они представляют собой зависимость истинно положительных и ложноположительных решений классификатора относительно значений параметров границ диапазона, а именно их изменения (рис. 3). Площадь (area) под ROC-кривой есть количественная интерпретация качества классификатора.

F-мера (F-score) сочетает в себе оценки точности и полноты, при этом остается чувствительной к распределению данных:

$$F\text{-}score = \frac{2 \cdot recall \cdot precision}{recall + precision}.$$

Характеристика обучающей выборки

Эффективность классификаторов, построенных с применением методов машинного обучения, во многом зависит от наличия качественного в смысле отсутствия зашумленности и наличия сбалансированности классов набора обучающих примеров — dataset. Популярными наборами данных в области построения классификаторов сетевых атак для IoT-устройств являются IoT-23, CTU-IoT-Malware-Capture, N_BaIoT и некоторые другие. Все эти обучающие примеры пожертвованы их создателями в виде услуги сообществу исследователей методов машинного обучения и имеются в открытом виде репозитория UCI.

Обучение классификаторов обнаружения атак на сенсорные сети выполнено на dataset N_BaIoT. Dataset содержит реальные данные трафика, собранные с девяти коммерческих IoT-устройств: дверного замка Danmini, термостата Ecobee, дверного замка Ennio, радионяни Philips_B120N10, камеры видеонаблюдения Provision PT737E, камеры видеонаблюдения Provision PT838, вебкамеры Samsung SNH1011N, камеры видеонаблюдения SimpleHome XCS71002WHT, камеры видеонаблюдения XCS71003WHT. Все устройства достоверно заражены Mirai и Bashlite.

Каждая запись набора данных представляет собой поведенческий снимок хостов и протоколов, по которым передавался пакет. Снимок получает контекст пакета путем извлечения 115 признаков, описание которых приведено в работе [7].

Наименования классов и количество примеров обучения dataset N_BaIoT приведены в табл. 2. Соотношение обучающих и тестовых примеров составило 80 и 20 % соответственно для каждого класса.

Для проводных сетей популярны следующие dataset.

KDDCup 1999 — dataset, включающий пять миллионов записей. Запись представляет собой 42 зафиксированных параметра, характеризующих пакеты, передаваемых по протоколам TCP, UDP и ICMP в определенные промежутки времени, при этом 41 параметр — это информационные признаки, а 42-й параметр — метка класса, обозначающая наименование атаки или ее отсут-

■ ***Таблица 2.*** Dataset N_BaIoT
■ ***Table 2.*** Dataset N_BaIoT

Название класса	Метка класса	Количество примеров
benign	0	98 514
combo_bashlite	1	18 339
junk_bashlite	2	9266
scan_bashlite	3	9052
tcp_bashlite	4	28 494
udp_bashlite	5	33 362
ack_mirai	6	13 307
scan_mirai	7	22 279
syn_mirai	8	14 192
udpplain_mirai	9	12 341
udp_mirai	10	36 393

ствие. В работе [20] приведено описание информационных признаков.

NSL-KDD 2009 — dataset, являющийся улучшением KDDCup 1999, в частности, не содержит дублирующих записей. Dataset NSL-KDD содержит 36 типов атак по четырем категориям:

1) Denial of Service (Dos) — атаки, ограничивающие доступ верифицированным пользователям к конкретному сервису через определенный протокол (Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm);

2) Remote to Local (R2L) — атаки, направленные на получение доступа к локальной машине пользователя из внешней среды (Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named);

3) User to Root (U2R) — атаки, направленные на получение привилегированных прав доступа к машине жертвы (Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps);

4) Probe — атаки, направленные на получение сведений об инфраструктуре пользователя (Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint) [20].

В целом NSL-KDD содержит 125 973 записи, предназначенные для обучения, и 22 544 записи для тестирования.

Кроме dataset KDDCup 1999 и его улучшенной версии NSL-KDD 2009, в области информационной безопасности инфокоммуникационных сетей существует много других обучающих выборок, например: ECML-PKDD 2007 — включает контекст web-ресурса, запроса и класс атаки; HTTPCSIC 2010 — включает несколько тысяч веб-запросов, которые предлагаются для тестирования систем защиты от веб-атак; ADFA2013 — содержит трафиковые трассы системных вызовов ОС Linux нормального режима работы и трассы сетевых атак; UNSW-NB15 2015 — содержит данные трафика, записанного в течение одного часа, насчитывающего девять типов атак, сгенерированных специальным программным обеспечением, и т. д.

Из множества перечисленных обучающих примеров для обнаружения атак в проводных сетях только NSL-KDD очищен от шумов, что позволяет использовать его сразу без предобработки, к тому же имеет наибольшую практику применения. Правда, классы в NSL-KDD несбалансированные, что вынуждает отказаться от классов, представленных малыми объемами примеров обучения. В табл. 3 приведены наименования классов и количество обучающих примеров классов, которые использованы в работе. Соотношение обучающих и тестовых примеров составило 80 и 20 % соответственно для каждого класса.

■ ***Таблица 3***. Dataset NSL-KDD после исключения классов малых объемов

■ ***Table 3***. Dataset NSL-KDD after excluding small volume classes

Название класса	Метка класса	Количество примеров
smurf	0	164 091
normal	1	60 593
neptune	2	58 001
snmpgetattack	3	7741
mailbomb	4	5000
guess_passwd	5	4367
snmpguess	6	2406
satan	7	1633
warezmaster	8	1602
back	9	1098
Mscan	10	1053
apache2	11	794
Processtable	12	759
Saint	13	736
Portsweep	14	354
Ipsweep	15	306
Httptunnel	16	158

Эксперименты и оценка результатов

Значения оценок достоверности методов машинного обучения по детектированию атак в сети IoT приведены в табл. 4.

Как видно из полученных результатов обучения:

— для беспроводной сенсорной сети IoT все методы машинного обучения показали высокую точность классификации. Это свидетельствует о качественном dataset;

— для проводной сети IoT случайный лес показывает для каждого класса высокие результаты, близкие к 100 % точности. В других методах есть «провалы» в распознавании определенных классов атак, например, для дерева решений — snmpgetattack с 65,27 % точности, для нейронной сети — portsweep с 61,70 % точности, для *k*-ближайших соседей — satan с 36,91 % и saint — с 13,12 % точности.

Результаты эффективности методов машинного обучения применительно к задаче классификации сетевых атак представлены на рис. 4–7.

Результаты сравнения методов машинного обучения по оценкам precision, recall и F-score идентичны в процентном соотношении оценке precision, приведенной на рис. 4.

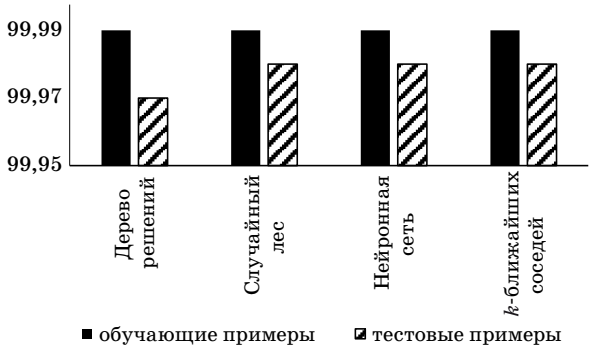
■ ***Таблица 4.*** Значения accuracy методов машинного обучения

■ ***Table 4.*** Values of accuracy of machine learning methods

Метка класса	Дерево решений	Случайный лес	Нейронная сеть	*k*-ближайших соседей
Беспроводная сенсорная сеть IoT				
benign	**100**	**100**	99,98	**100**
combo_ bashlite	99,98	99, 74	**100**	99,99
junk_bashlite	99,88	99,81	99,92	99,96
scan_bashlite	99,96	**100**	99,92	99,98
tcp_bashlite	99,97	99,97	99,97	99,99
udp_bashlite	99,98	**100**	99,96	99,99
ack_mirai	99,97	99,97	99,97	99,99
scan_mirai	**100**	**100**	**100**	**100**
syn_mirai	**100**	**100**	**100**	**100**
udpplain_ mirai	**100**	**100**	**100**	**100**
Проводная вычислительная сеть IoT				
smurf	**100**	**100**	99,5	**100**
normal	94,64	94,75	92	93,52
neptune	99,99	**100**	99,9	99,76
snmpgetattack	65,27	**100**	99,9	70,18
mailbomb	99,93	95,19	94,9	**100,00**
guess_passwd	99,51	99,93	97,0	99,22
snmpguess	99,71	**100**	97,9	**100,00**
satan	97,53	97,11	81,8	36,91
warezmaster	99,36	98,73	98,1	99,36
back	99,71	**100**	97,7	99,14
mscan	99,08	**100**	98,1	99,39
apache2	**100**	**100**	99,7	96,34
processtable	**100**	**100**	98,4	**100**
saint	91,40	92,76	**100,0**	13,12
portsweep	**100**	**100**	61,7	96,19
ipsweep	**100**	**100**	79,3	97,78
httptunnel	**100**	**100**	**100,0**	92,73

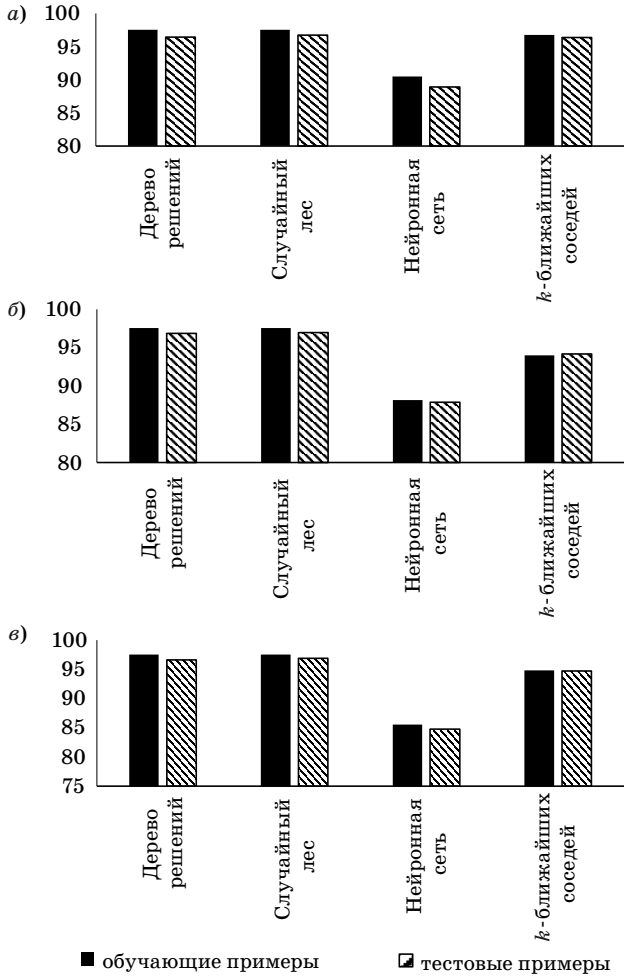
Результаты ROC-кривых для других классификаторов, обученных методами случайный лес, нейронная сеть и *k*-ближайших соседей для беспроводной сенсорной сети IoT, идентичны ROC-кривой, приведенной на рис. 6.

ROC-кривые для классификаторов СОА на уровне проводной локальной или глобальной сети представлены на рис. 7, *а–г*.



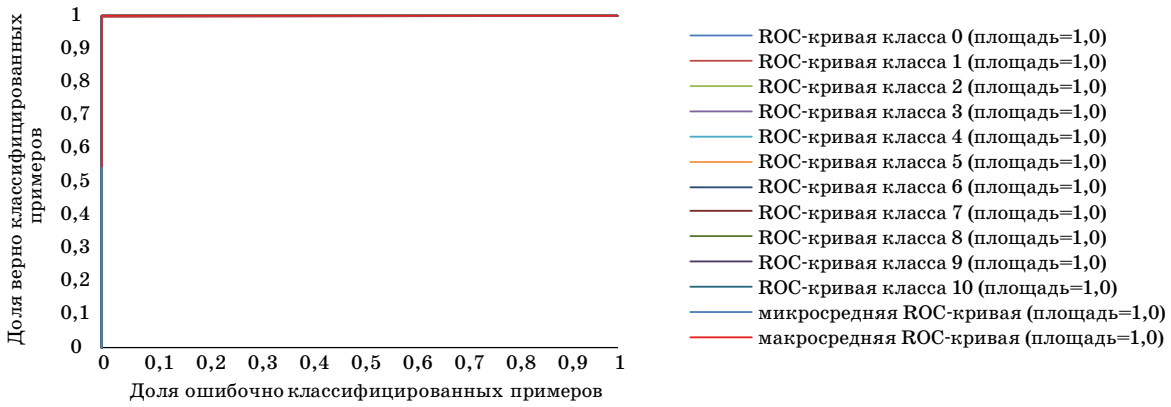
■ ***Рис. 4.*** Результаты оценки precision методов машинного обучения при классификации сетевых атак сенсорной сети

■ ***Fig. 4.*** Results of precision evaluation of machine learning methods in classifying sensor network attacks



■ ***Рис. 5.*** Результаты оценки эффективности методов машинного обучения при классификации сетевых атак в проводной сети: *а* — precision; *б* — recall; *в* — F-score

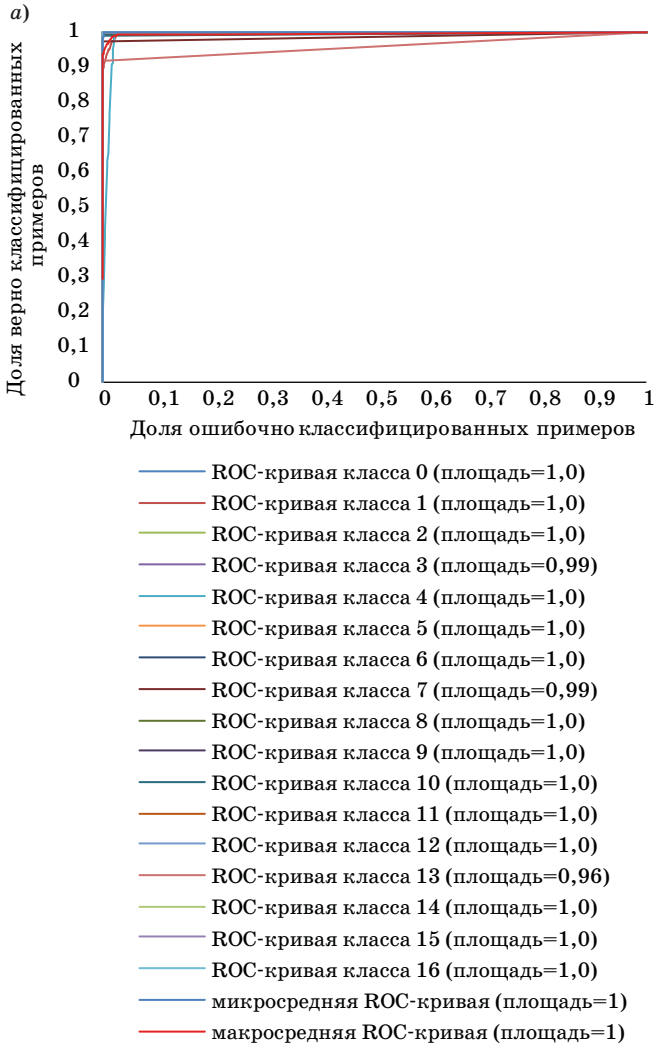
■ ***Fig. 5.*** Results of evaluating the effectiveness of machine learning methods in the classification of network attacks in a wired network: *а* — precision; *б* — recall; *в* — F-score



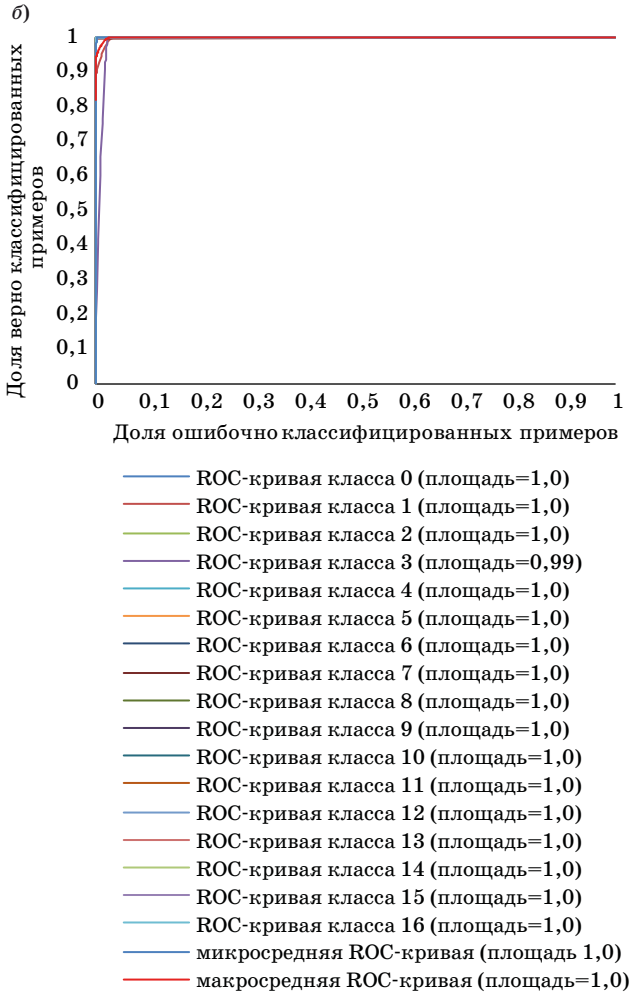
■ ***Рис. 6.*** ROC-кривые, полученные методом дерево решений для беспроводной сенсорной сети

■ ***Fig. 6.*** ROC-curves obtained by the method decision tree for a wireless sensor network

а)

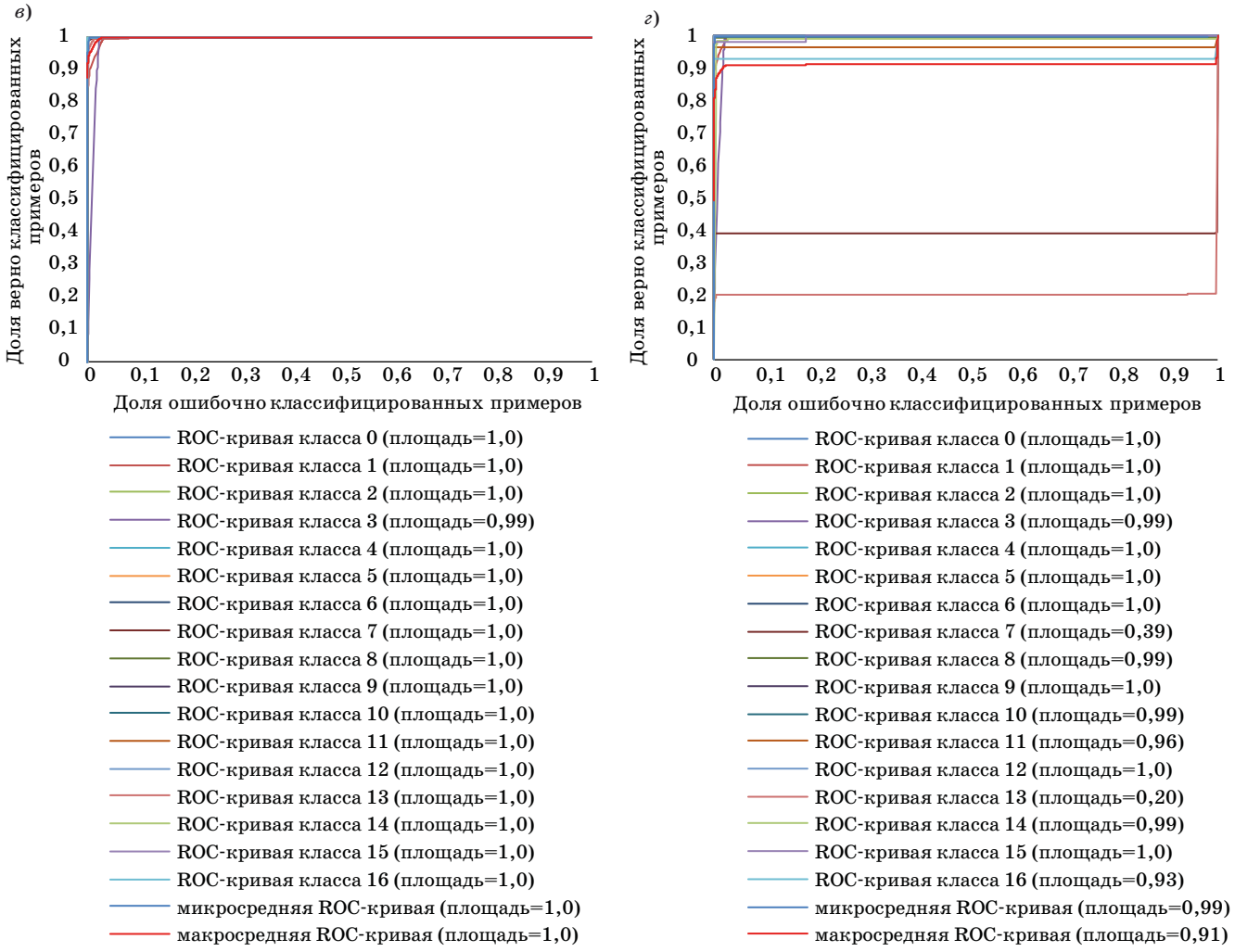


б)



■ ***Рис. 7.*** ROC-кривые, полученные методами дерево решений (*а*); случайный лес (*б*); нейронная сеть (*в*); *k*-ближайших соседей (*г*) для проводной сети IoT (см. также с. 49)

■ ***Fig. 7.*** ROC-curves obtained by the method: decision tree (*а*); random forest (*б*); neural network (*в*); *k*-nearest neighbors (*г*) for wired IoT network (see also p. 49)



■ ***Окончание рис. 7***
■ ***Ending of Fig. 7***

Приведенные на рис. 7, *а–г* ROC-кривые показывают высокую эффективность обученных классификаторов. Так, для методов дерево решений, случайный лес и нейронная сеть значения отношений истинно положительных и ложноположительных решений классификаторов близки к единице. Для метода *k*-ближайших соседей не могут быть достоверно обнаружены атаки satan и saint.

Содержание этапов предлагаемого подхода к обнаружению атак в сетях интернета вещей

Предлагаемый подход к обнаружению атак в сетях интернета вещей включает следующие основные этапы:

1) сбор статистики о передаваемом трафике;

2) извлечение признаков из собранной статистики;

3) классификация трафика;

4) непрерывный мониторинг сети IoT.

На этапе сбора данных осуществляется съем статистик с:

— головных узлов кластеров беспроводной сенсорной сети, принимающих данные от IoT-устройств;

— маршрутизаторов и шлюзов проводной локальной и глобальной сети;

— протоколов передачи данных.

Узловые и сетевые агенты доставляют собранные статистики в СОА.

На втором этапе из собранных статистик извлекаются признаки отдельно для классификатора беспроводной сенсорной сети и для проводной локальной и (или) глобальной сети. Признаки те же самые, что применялись для обучения классификаторов.

На третьем этапе извлеченные признаки подаются на входы соответствующих классификато-

ров. Обнаруженные аномалии в трафике, передаваемом от IoT-устройств, могут указывать на то, что устройство скомпрометировано. Признаки, указывающие на аномальное поведение какого-либо сенсорного устройства, передаются в блок принятия решений СОА, где происходит анализ признаков и принимается решение о реагировании на сложившуюся ситуацию. Атаки, обнаруженные на уровне проводной локальной или глобальной сети, также передаются в блок принятия решений СОА для выбора вариантов реагирования на выявленную атаку или класс атак.

Непрерывный мониторинг подразумевает выполнение этапов 1–3 с периодичностью, которая может совпадать, например, с продолжительностью раундов, во время которых сенсорные устройства передают данные на головной узел своего кластера беспроводной сенсорной сети.

При регистрации нового IoT-устройства трафик, генерируемый им, должен быть собран сразу после подключения IoT-устройства к сети, чтобы гарантировать, что данные являются незараженными. Таким образом формируется база данных профилей нормального поведения каждого IoT-устройства.

При появлении нового вида или класса сетевых атак рассматриваемый подход подразумевает организацию сбора статистик зараженного трафика в целях дальнейшего обучения классификатора на обнаружение этих атак. Извлеченные признаки представляют собой профиль конкретного вида или класса атаки.

Таким образом, на головных узлах кластеров беспроводной сенсорной сети функционирует модуль обнаружения аномального поведения IoT-устройств, представляющий собой обученный классификатор; на маршрутизаторах и шлюзах функционирует модуль обнаружения сетевых атак, также представляющий собой обученный классификатор.

Заключение

С ростом объемов цифровых данных в методах выявления атак стали актуальны исследования, связанные с применением методов машинного обучения для обнаружения аномалий сетевого трафика — наличия сетевых атак. Это в полной мере относится и к сетям интернета вещей, а постоянное подключение сенсорных устройств к интернету делает их удобным инструментом для организации кибератак.

На уровне беспроводной сенсорной сети обнаружение аномального поведения IoT-устройств реализуется оценкой отклонения поведения IoT-устройств от соответствующих им профилей поведения. Построение профилей умных устройств осуществляется на основе статистических характеристик, снимаемых с узлов сети интернета вещей и протоколов, по которым передаются пакеты данных. На уровне локальной или глобальной проводной сети интернета вещей происходит агрегирование данных, анализ которых также выполняется методами машинного обучения.

Обученные модели классификации могут стать частью системы обнаружения сетевых атак, принимающих решение о компрометации узла «на лету».

Экспериментальным путем выбрана модель классификатора сетевых атак на уровне беспроводной сенсорной сети и локальной или глобальной проводной сети. Лучшие результаты в смысле оценок полноты и точности продемонстрированы методом случайного леса для проводной локальной или глобальной сети и всеми рассмотренными методами для беспроводной сенсорной сети.

На качество классификаторов существенное влияние оказывает наличие сбалансированного и подготовленного набора данных, что само по себе является трудоемкой работой. В дальнейшем планируется систематизировать информационные параметры, которые могут иметь наибольшую важность при обучении классификаторов. А также выполнить сравнение двух подходов к построению СОА: основанных на методах машинного обучения и методах оценки доверия узла к своим соседям.

Литература

1. **Киричек Р. В., Парамонов А. И., Прокопьев А. В., Кучерявый А. Е.** Эволюция исследований в области беспроводных сенсорных сетей. *Информационные технологии и телекоммуникации*, 2014, № 4 (8), с. 29–41. http://ijitt.ru/ (дата обращения: 22.08.2021).
2. **Шелухин О. И., Сакалема Д. Ж., Филинова А. С.** Обнаружение вторжений в компьютерные сети. Сетевые аномалии. М., Горячая линия — Телеком, 2013. 220 с.
3. **Татарникова Т. М., Богданов П. Ю., Краева Е. В.** Предложения по обеспечению безопасности системы умного дома, основанные на оценке потребляемых ресурсов. *Проблемы информационной безопасности. Компьютерные системы*, 2020, № 4, с. 88–94.
4. **Baddar S. A.-H., Merlo A., Megiliardi M.** Anomaly detection in computer networks: A state-of-the-art review. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 2014, vol. 5, no. 4, pp. 29–64.
5. **Колбанёв М. О., Пойманова Е. Д., Татарникова Т. М.** Физические ресурсы информационного процесса со-

хранения данных. *Известия высших учебных заведений. Приборостроение*, 2014, т. 57, № 9, с. 38–42.
6. **Lee P.** *Internet of Things for Architects*. Birmingham — Mumbai, Packt Publ., 2018. 524 p.
7. **Meidan Y., Bohadana M., Mathov Y., Mirsky Y., Breitenbacher D., Shabtai A., and Elovici Y.** N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing, Special Issue — Securing the IoT*, 2018, vol. 17(3), pp. 12–22.
8. **Лоднева О. Н., Ромасевич Е. П.** Анализ трафика устройств интернета вещей. *Современные информационные технологии и IT-образование*, 2018, т. 14, № 1, с. 149–169. doi:10.25559/SITITO.14.201801.149-169
9. **Kumar S., Spafford E. H.** A pattern matching model for misuse intrusion detection. *Proceedings of the 17th National Computer Security Conference*, 1994, pp. 11–21.
10. **Thatte G., Mitra U., Heidemann J.** Parametric methods for anomaly detection in aggregate traffic. *IEEE/ACM Transaction on Networking*, 2011, vol. 19(2), pp. 512–525.
11. **Викулов А. С., Парамонов А. И.** Анализ трафика в сети беспроводного доступа стандарта IEEE 802.11. *Труды учебных заведений связи*, 2017, т. 3, № 3, с. 21–27.
12. **Wu S. X., Banzhaf W.** The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing*, 2010, vol. 10(1), pp. 1–35.
13. **Браницкий А. А., Котенко И. В.** Анализ и классификация методов обнаружения сетевых атак. *Труды СПИИРАН*, 2016, вып. 2(45), с. 207–244. doi:10.15622/SP.45.13
14. **Ingre B., Yadav A., Soni A. K.** Decision tree based intrusion detection system for NSL-KDD dataset. *Proceedings of the International Conference on Information and Communication Technology for Intelligent Systems (ICTIS)*, Ahmedabad, India, March 25–26, 2017, Cham, Springer, 2017, vol. 2, pp. 207–218. doi:10.1007/978-3-319-63645-0_23
15. **Fatihand E., Aydin G.** Data classification with deep learning using tensorflow. *International Conference on Computer Science and Engineering*, 2017, pp. 755–758.
16. **Татарникова Т. М.** Ограничения утечки информации посредством неочевидных функций смартфона Android 5. *Информационно-управляющие системы*, 2019, № 5, с. 24–29. doi:10.31799/16848853-2019-5-24-29
17. **Gyanchandani M., Rana J. L., Yadav R. N.** Taxonomy of anomal based intrusion detection system: A review. *International Journal of Scientific and Research Publications*, 2012, vol. 2(12), pp. 1–13.
18. **Татарникова Т. М., Бимбетов Ф., Богданов П. Ю.** Выявление аномалий сетевого трафика методом глубокого обучения. *Известия ЛЭТИ*, 2021, № 4, с. 36–41.
19. **Jyothsna V., Prasad V. V. R.** A Review of anomaly based intrusion detection systems. *International Journal of Computer Applications*, 2011, vol. 28, no. 7, pp. 26–35.
20. *A Deeper Dive into the NSL-KDD Data Set — Towards Data Science*. https://towardsdatascience.com/a-deeper-dive-into-the-nsl-kdd-data-set-15c753364657 (дата обращения: 18.09.2021).

UDC 004.07
doi:10.31799/1684-8853-2021-6-42-52

Intrusion detection in internet of things networks based on machine learning methods

T. M. Tatarnikova[a], Dr. Sc., Tech., Professor, orcid.org/0000-0002-6419-0072, tm-tatarn@yandex.ru
P. Yu. Bogdanov[a], Senior Lecturer, orcid.org/0000-0002-7533-7316
[a]Saint-Petersburg State University of Aerospace Instrumentation, 67, B. Morskaia St., 190000, Saint-Petersburg, Russian Federation

Introduction: The growing amount of digital data generated, among others, by smart devices of the Internet of Things makes it important to study the application of machine learning methods to the detection of network traffic anomalies, namely the presence of network attacks. **Purpose:** To propose a unified approach to detecting attacks at different levels of IoT network architecture, based on machine learning methods. **Results:** It was shown that at the wireless sensor network level, attack detection is associated with the detection of anomalous behavior of IoT devices, when the deviation of an IoT device behavior from its profile exceeds a predetermined level. Smart IoT devices are profiled on the basis of statistical characteristics, such as the intensity and duration of packet transmission, the proportion of retransmitted packets, etc. At the level of a local or global wired IoT network, data is aggregated and then analyzed using machine learning methods. Trained classifiers can become a part of a network attack detection system, making decisions about compromising a node on the fly. Models of classifiers of network attacks were experimentally selected both at the level of a wireless sensor network and at the level of a local or global wired network. The best results in terms of completeness and accuracy estimates are demonstrated by the random forest method for a wired local and/or global network and by all the considered methods for a wireless sensor network. **Practical relevance:** The proposed models of classifiers can be used for developing intrusion detection systems in IoT networks.

Keywords — network attack, Internet of Things, training sample, intrusion detection system, classifier model efficiency.

For citation: Tatarnikova T. M., Bogdanov P. Yu. Intrusion detection in internet of things networks based on machine learning methods. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 6, pp. 42–52 (In Russian). doi:10.31799/1684-8853-2021-6-42-52

References

1. Kirichek R. V., Paramonov A. I., Prokopiev A. V., Koucheryavy A. E. The investigation evolution in the wireless sensor networks area. *Telecom IT*, 2014, vol. 8, no. 4, pp. 29–41. Available at: http://ijitt.ru/ (accessed 22 September 2021) (In Russian).
2. Sheluhin O. I., Sakalema D. Zh., Filinova A. S. *Obnaruzheniye vtorzheniy v komp'yuternyye seti. Setevye anomalii* [Intrusion detection in computer networks. Network anomalies]. Moscow, Goriachaia liniia — Telekom Publ., 2013. 220 p. (In Russian).
3. Tatarnikova T. M., Bogdanov P. Yu., Kraeva E. V. Smart home security proposals based on assessment of consumption resources. *Problems of Information Security. Computer Systems*, 2020, no 4, pp. 88–94 (In Russian).
4. Baddar S. A.-H., Merlo A., Migliardi M. Anomaly detection in computer networks: A state-of-the-art review. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 2014, vol. 5, no. 4, pp. 29–64.
5. Kolbanev M. O., Poimanova E. D., Tatarnikova T. M. Physical resources of the information process of data saving. *Journal of Instrument Engineering*, 2014, vol. 57, no. 9, pp. 38–42 (In Russian).
6. Lee P. *Internet of Things for Architects*. Packt Publ., Birmingham — Mumbai, 2018. 524 p.
7. Meidan Y., Bohadana M., Mathov Y., Mirsky Y., Breitenbacher D., Shabtai A., and Elovici Y. N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing, Special Issue — Securing the IoT*, 2018, vol. 17, no. 3, pp. 12–22.
8. Lodneva O. N., Romasevich E. P. Analysis of devices traffic of the internet of things. *Modern Information Technologies and IT-Education*, 2018, vol. 14, no. 1, pp. 149–169. doi:10.25559/SITITO.14.201801.149-169 (In Russian).
9. Kumar S., Spafford E. H. A pattern matching model for misuse intrusion detection. *Proceedings of the 17th National Computer Security Conference*, 1994, pp. 11–21.
10. Thatte G., Mitra U., Heidemann J. Parametric methods for anomaly detection in aggregate traffic. *IEEE / ACM Transaction on Networking*, 2011, vol. 19(2), pp. 512–525.
11. Vikulov A., Paramonov A. IEEE 802.11 WLAN traffic analysis. *Proceedings of Telecommunication Universities*, 2017, vol. 3, iss. 3, pp. 21–27 (In Russian).
12. Wu S. X., Banzhaf W. The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing*, 2010, vol. 10(1), pp. 1–35.
13. Branitskiy A. A., Kotenko I. V. Analysis and classification of methods for network attack detection. *SPIIPAS Proceedings*, 2016, iss. 2 (45), pp. 207–244. doi:10.15622/SP.45.13 (In Russian).
14. Ingre B., Yadav A., Soni A. K. Decision tree based intrusion detection system for NSL-KDD dataset. *Proceedings of the International Conference on Information and Communication Technology for Intelligent Systems*, *(ICTIS*), Ahmedabad, India, March 25–26, 2017, Cham, Springer, 2017, vol. 2, pp. 207–218. doi:10.1007/978-3-319-63645-0_23
15. Fatihand E., Aydin G. Data classification with deep learning using tensorflow. *International Conference on Computer Science and Engineering*, 2017, pp. 755–758.
16. Tatarnikova T. M. Restricting data leakage through non-obvious features of Android 5 smartphone. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2019, no. 5, pp. 24–29 (In Russian). doi:10.31799/16848853-2019-5-24-29
17. Gyanchandani M., Rana J. L., Yadav R. N. Taxonomy of anomal based intrusion detection system: A review. *International Journal of Scientific and Research Publications*, 2012, vol. 2(12), pp. 1–13.
18. Tatarnikova T. M., Bimbetov F., Bogdanov P. Yu. Identifying network traffic anomalies by deep learning. *Proceedings of Saint Petersburg Electrotechnical University Journal*, 2021, no. 4, pp. 36–41 (In Russian).
19. Jyothsna V., Prasad V. V. R. Review of anomaly вased intrusion detection systems. *International Journal of Computer Applications*, 2011, vol. 28, no. 7, pp. 26–35.
20. *A Deeper Dive into the NSL-KDD Data Set — Towards Data Science*. Available at: https://towardsdatascience.com/a-deeper-dive-into-the-nsl-kdd-data-set-15c753364657 (accessed 18 September 2021).