ОБНАРУЖЕНИЕ ВТОРЖЕНИЙ НА ОСНОВЕ ФЕДЕРАТИВНОГО ОБУЧЕНИЯ: АРХИТЕКТУРА СИСТЕМЫ И ЭКСПЕРИМЕНТЫ

Новикова Е.С.[1], Котенко И.В.[2], Мелешко А.В.[3], Израилов К.Е.[4]

Цель исследования: *разработка подхода к построению системы обнаружения вторжений на основе федеративного машинного обучения.*

Полученный результат: *разработана концепция и архитектура системы обнаружения вторжений на основе федеративного машинного обучения. Предложенная архитектура включает новые компоненты, отвечающие за организацию федеративного обучения, такие как компоненты выбора данных, обучения локальной модели, оценки рисков конфиденциальной информации, выявления атак на федеративное обучение, а также определяет их связи с другими функциональным элементами системы. Для выполнения экспериментальной оценки компонентов системы обнаружения вторжений на основе федеративного обучения сформулированы метрики оценки их эффективности, которые позволяют оценить в том числе требования к вычислительным ресурсам системы. Предложен подход к моделированию распределения данных между взаимодействующими компонентами, и получены экспериментальные оценки эффективности обнаружения вторжений с помощью моделей машинного обучения, обученных в федеративном режиме.*

Научная новизна: *анализ литературы показал, что применение федеративного обучения для построения систем обнаружения вторжений связано с рядом открытых практических задач; в частности, отсутствует общая методология построения и оценки эффективности таких систем. В настоящей работе предлагается архитектура системы обнаружения вторжений, которая учитывает практические особенности использования федеративного обучения, а также представляются результаты экспериментальной оценки эффективности применения моделей обнаружения вторжений, обученных в федеративном режиме.*

Вклад: *Новикова Е. С. и Котенко И. В. — общая концепция построения и архитектура системы обнаружения вторжения с использованием федеративного машинного обучения, методология сбора данных для исследования безопасности киберфизических систем; Новикова Е. С. и Израилов К. Е. — проработка функциональности отдельных компоненты системы обнаружения вторжения, Мелешко А. В. — проведение экспериментов.*

Ключевые слова: *кибербезопасность, киберфизические системы, выявление аномалий и кибератак, распределенное машинное обучение, сверточная нейронная сеть, оценка эффективности.*

DOI: 10.21681/2311-3456-2023-6-50-66

Введение

В настоящее время предложены разнообразные подходы к обнаружению вторжений и аномалий в компьютерных сетях [1-3]. В их основе лежат методы на основе сигнатурного анализа, статистического анализа временных рядов [4], а также алгоритмы на основе методов классического машинного обучения (МО) [5] и глубокого обучения [6-7]. На практике наибольшее распространение получили методы на осно-

1 Новикова Евгения Сергеевна, кандидат технических наук, доцент, старший научный сотрудник лаборатории проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук (СПб ФИЦ РАН), г. Санкт-Петербург, Россия. ORCID: 0000-0003-2923-4954. Scopus Author ID: 55415626100. E-mail: novikova@comsec.spb.ru

2 Котенко Игорь Витальевич, заслуженный деятель науки РФ, доктор технических наук, профессор, главный научный сотрудник и руководитель лаборатории проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук, Санкт-Петербург. ORCID: 0000-0001-6859-7120. Scopus Author ID: 15925268000. E-mail: ivkote@comsec.spb.ru.

3 Мелешко Алексей Викторович, младший научный сотрудник лаборатории проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук, Санкт-Петербург. ORCID: 0000-0002-1209-4230. Scopus Author ID: 57214672771. E-mail: meleshko.a@iias.spb.su.

4 Израилов Константин Евгеньевич, кандидат технических наук, доцент, старший научный сотрудник лаборатории проблем компьютерной безопасности Санкт-Петербургского Федерального исследовательского центра Российской академии наук, Санкт-Петербург. ORCID: 0000-0002-9412-5693. Scopus Author ID: 56123238800. E-mail: konstantin.izrailov@mail.ru.

ве правил и сигнатур в силу их простоты внедрения и прозрачности получаемых результатов. Ключевой преградой к широкому практическому применению подходов на основе МО является необходимость их адаптации (переобучения) к защищаемой системе, что является ресурсоемким и трудоемким процессом, который требует наличия хорошо структурированных размеченных наборов данных. Доступ к таким данным часто ограничен, что также значительно затрудняет процесс тестирования и внедрения компонент обнаружения вторжения на основе МО, несмотря на их способность обнаруживать сложные, многошаговые и растянутые во времени атаки.

Одним из заметных достижений в области МО в последнее время стало определение концепции федеративного обучения (ФО) как способа организации распределенного МО, при котором владельцы данных не обязаны делиться ими для построения модели МО [8]. Формирование глобальной модели осуществляется итеративно на основе обновлений, полученных от узлов — владельцев данных. Такая распределенная схема позволяет строить аналитические системы [9], в основе которых лежит МО, при этом сохраняя конфиденциальность данных конечных пользователей. Кроме того, она дает возможность естественным образом расширить обучающую выборку.

В кибербезопасности ФО может быть рассмотрено как механизм обмена данными об угрозах и атаках на защищаемые системы без необходимости распространения реальных данных, способствуя тем самым развитию совместных подходов к реализации и построению эффективных систем обнаружения и противодействия кибератакам. Несмотря на то, что в последнее время предложено большое число подходов к обнаружению вторжений на основе ФО, многие практические вопросы по его использованию остаются открытыми [10-13]: например, построение системы обнаружения вторжений (СОВ) на основе ФО, оценивание эффективности обнаружения вторжений, определение требований к вычислительным ресурсам и пропускной способности канала связи компонентов (что особенно важно для систем на базе технологии Интернета Вещей).

В настоящей работе представлена архитектура СОВ на основе ФО и даны описания основных ее компонентов. Для оценки эффективности применения ФО в СОВ и определения вычислительных требований к компоненту СОВ на его основе предложен сценарий эксперимента, который определяет схему распределения данных между клиентами и метрики оценки эффективности моделей МО, обученных в федеративном режиме.

Статья структурирована следующим образом. В разделе 1 кратко представлена концепция ФО, а в разделе 2 дается анализ релевантных работ. В разделе 3 обсуждается архитектура СОВ и приводится описание ее компонентов, выполняющих ФО, в разделе 4 представлены описание экспериментов и полученные результаты. В разделе 5 делаются выводы и формулируются направления дальнейших работ.

1. Федеративное обучение

ФО является способом организации распределенного МО, при котором данные не собираются в единое централизованное хранилище, а используются для выполнения локального обучения на узлах их генерации; для формирования общей или глобальной аналитической модели результаты локального обучения объединяются (агрегируются) специальным образом, который зависит от модели. Таким образом, составными элементами ФО являются следующие компоненты:

1) клиенты — узлы, которые генерируют и накапливают данные, а также выполняют обучение локальных моделей;

2) агрегирующий сервер — узел, который управляет процессом обучения в федеративном режиме и вычисляет глобальную модель;

3) коммуникационно-вычислительная среда — сетевое пространство, обеспечивающее передачу информации между клиентами и сервером.

Формально, ФО определяется следующем образом. Пусть $C = \{c_i\}_{i=0}^{n}$ –множество из n клиентов, каждый из которых владеет некоторым набором данных d_i; при этом, клиенты желают совместно обучить некоторую аналитическую модель на всем множестве наборов данных. В случае традиционного МО все наборы данных d_i объединяются в единый набор $D = \{d_0 \cup d_1 \cup \ldots \cup d_n\}$, на котором обучается модель M_D с некоторой точностью $A(M_D)$. В случае ФО множество наборов данных не объединяется, а глобальная модель M_{FL} вычисляется на основе локально обученных моделей M_{di}, причем ее точность $A(M_{FL})$ должна удовлетворять следующему требованию:

$$|A(M_D) - A(M_{FL})| \leq \delta,$$

где δ — неотрицательное вещественное число, т.е. разница в точности этих моделей не должна превышать некоторый заданный порог δ.

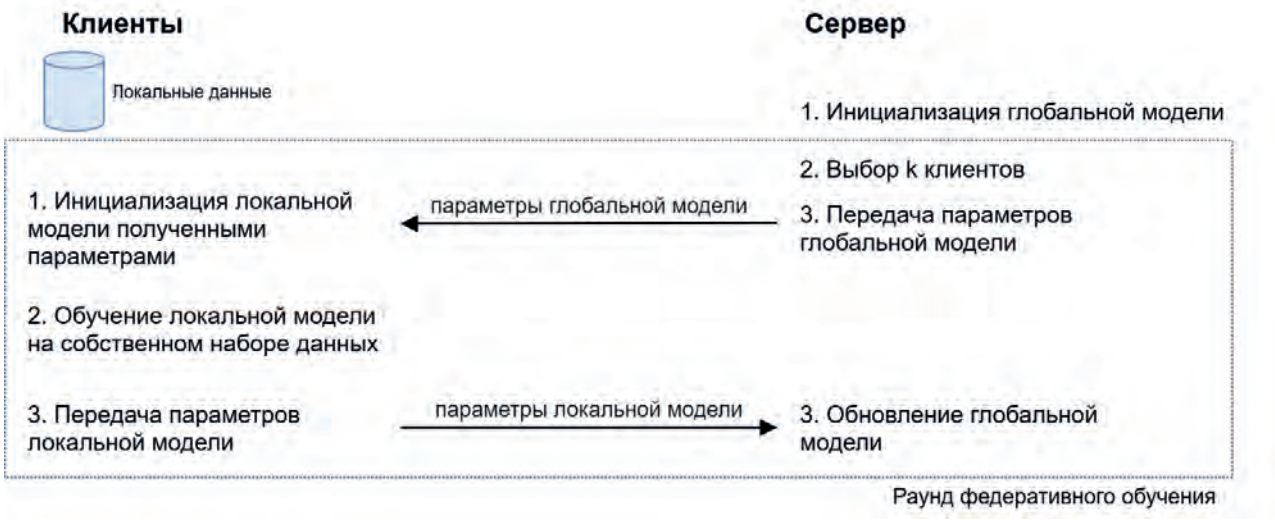

Рис. 1. Упрощенная схема раунда федеративного обучения

Процесс ФО состоит из нескольких шагов, которые выполняются итеративно. В начале каждого раунда (т. е. итерации) из *n* клиентов случайным образом выбирается *k* клиентов, которым агрегирующий сервер пересылает текущие параметры глобальной модели. Затем каждый отобранный клиент выполняет обучение локальной модели на собственном наборе данных, а полученные результаты отправляет агрегирующему серверу. Последний, получив новые данные от клиентов, обновляет глобальную модель, и процесс обучения повторяется. Упрощенная схема раунда ФО представлена на рис. 1.

В настоящее время предложены варианты протоколов ФО, которые обеспечивают аутентификацию клиентов и агрегирующего сервера, а также подтверждение подлинности источника передаваемых параметров модели [14, 15].

Наиболее часто используемым алгоритмом агрегирования для формирования глобальной модели является *федеративное усреднение* (*FederatedAveraging*) [8], который основан на определении весовой суммы параметров локальных моделей, чьи веса пропорциональны размеру соответствующей обучающей выборки клиента.

Системы ФО обычно характеризуются тремя следующими свойствами: схемой взаимодействия между клиентами, схемой разделения данных, а также вычислительными и сетевыми ресурсами, доступными взаимодействующим клиентам.

Схема взаимодействия между клиентами определяет то, каким образом осуществляется координация процесса ФО в целом, и какой участник отвечает за формирование глобальной модели — т.е. кто выполняет функции агрегирующего сервера. В централизованной схеме ФО выделяется отдельный узел, выполняющий роль сервера-агрегатора. Другие участники ФО передают параметры локальных моделей данному узлу и, соответственно, получают от него обновления глобальной модели. В случае децентрализованной схемы ФО (также известной как роевое обучение), функции агрегирующего сервера распределены между всеми участниками процесса, а для формирования глобальной модели результаты локального обучения рассылаются всем участникам.

Схема разделения данных определяет распределение атрибутов и объектов в наборах данных, принадлежащих разным клиентам. Пусть набор данных *DS* задается парой множеств $DS = \langle E, A \rangle$, где E — это множество объектов (например, сетевых потоков), а A — множество атрибутов, характеризующих эти объекты (например, длительность потока, число переданных байт или пакетов, количество соединений и т. п.). Выделяют два основных способа разделения данных между k клиентами — горизонтальный и вертикальный.

В первом случае каждый *i*-й клиент владеет собственным набором данных DS_i, полученных путем выделения подмножества объектов:

$$\begin{cases} \forall 1 \geq i \geq k : DS_i = \langle E_i, A \rangle \\ E = \{E_1 \cup E_2 \cup \ldots \cup E_k\} \\ \forall 1 \geq j \geq k, j \neq i : E_j \cap E_i = \emptyset \end{cases}.$$

Во втором случае каждый j-й клиент владеет собственным набором данных DS_j, полученных путем выделения подмножества атрибутов:

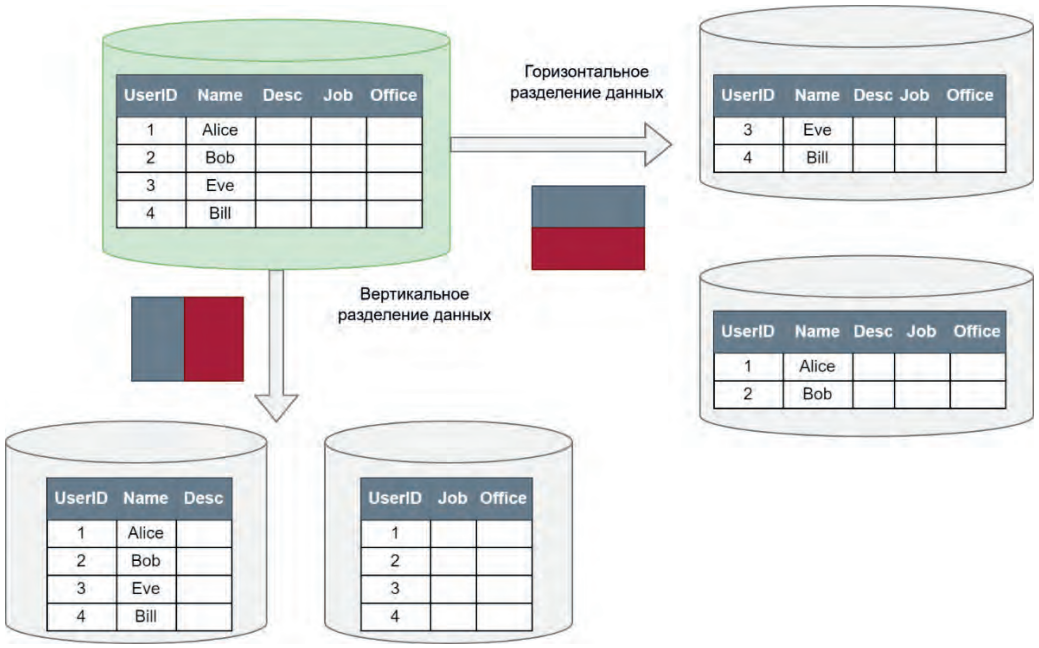

Рис. 2. Схема разделения данных

$$\begin{cases} \forall 1 \geq j \geq k: DS_j = \langle E, A_j \rangle \\ A = \{A_1 \cup A_2 \cup \ldots \cup A_k\} \\ \forall 1 \geq i \geq k, i \neq j: A_i \cap A_j = \emptyset \end{cases}.$$

На рис. 2 показаны примеры для указанных схем разделения данных между двумя клиентами.

Исходя из характеристик вычислительных и сетевых ресурсов, которые доступны взаимодействующим клиентам, обычно выделяют федерацию устройств и федерацию организаций. Для федерации устройств характерны (1) большое число возможных участников ФО с достаточно ограниченными вычислительными ресурсами и (2) нестабильное подключение устройств во время обучения; например, объединение мобильных устройств связи. Для федерации организации напротив характерно небольшое число участников с достаточно мощными вычислительными ресурсами, большим каналом связи и стабильным подключением во время обучения; например, крупные компании и организации.

2. Анализ релевантных работ

Исследование работ, посвященных построению систем обнаружения вторжений на основе ФО, показал, что чаще всего для построения СОВ используется централизованная топология ФО с горизонтальным разделением атрибутов [16-18]. Например, в [19] предложен подход к обнаружению вторжений для систем, построенных на основе технологии Интернета вещей. Авторы используют рекуррентную нейронную сеть, обученную в федеративном режиме для выявления аномалий в поведении устройств заданного типа. Построение аналитических моделей для каждого типа устройств позволяет задать горизонтальную схему разделения данных. В качестве тестового набора данных был использован набор данных, собранный с помощью тестового стенда, разработанного авторами [19] и состоящего из 14 устройств «умного» дома.

Похожая задача решается в [20], однако в этой работе в качестве тестового набора данных был использован набор N-BaIoT [21], который моделирует сетевой трафик от 9 реальных устройств Интернета вещей разного типа. Исходный набор данных был разделен между тремя клиентами по 100 000 записей на каждом, а соотношение нормальных и аномальных записей в каждом локальном наборе в проводимых экспериментах варьировалось. Основной акцент в работе был сделан на оценку влияния архитектуры нейронной сети на эффективность выявления аномалий и вторжений в различных сценариях распределения данных.

Система обнаружения FELIDS представлена в [13]. Для моделирования взаимодействия различных агентов авторы использовали несколько наборов данных: CSE-CIC-IDS2018 [22], MQTTset [23], и InSDN [24], а для организации ФО была использована специализированная программная библиотека Sherpa.ai [25]. Авторы тестировали эффективность нейронных сетей с различной архитектурой, обученных в федеративном режиме, и показали, что наиболее эффективной является полносвязная сеть, точность обнаружения атак которой достигла 98.54%.

В [26] предложена двухуровневая иерархическая распределенная СОВ на основе ФО. Причиной такого решения является высокий уровень неоднородности информационных технологий и подходов, используемых для построения информационных систем различных организаций. Авторы предложили разделить подсети отдельных организации на сегменты, и соответственно, процесс ФО выполнялся сначала на уровне сегмента — локальный или промежуточный уровень, а на глобальном уровне модели промежуточного уровня были объединены в единую глобальную модель. Для выполнения экспериментов был использован набор BoT-IoT [27]. Особенностью данной работы является использование технологии блокчейна для обеспечения неизменности результатов промежуточного и глобального обучения.

Децентрализованная схема ФО предложена для построения СОВ, разрабатываемых для интеллектуальных транспортных систем [28-30]. Данное решение обусловлено, в первую очередь, географической распределенностью таких систем и разнообразием транспортных маршрутов. В этом случае также используется иерархическая двухуровневая модель ФО — на нижнем уровне транспортные средства собирают данные вокруг себя и обновляют модели, полученные от базовых станций. Последние, с помощью технологии блокчейн фиксируют и валидируют все обновления, и после опубликования локальных моделей они уже выполняют формирование общей глобальной модели. Следует отметить, что эксперименты в [28] проводились при помощи двух открытых наборов данных — Car-Hacking, TON_IoT, близких к исследуемой предметной области. В [30] эксперименты были выполнены на таких наборах данных, как MNIST и CIFAR, которые представляют собой коллекцию изображений, что не позволяет судить о применимости результатов к задаче обнаружения вторжений.

Случай вертикально разделенных данных гораздо сложнее и практически не исследован в задачах обнаружения вторжений [31].

Таким образом, можно заключить, что на текущий момент все опубликованные подходы в предметной области представляют собой простой анализ применимости ФО к обнаружению аномалий и вторжений, который заключается в использовании современного и актуального набора данных, моделировании его распределения по множеству взаимодействующих клиентов и оценке точности обнаружения. Вопросы, связанные с оценкой требуемых вычислительных ресурсов для СОВ на основе ФО, практически не решаются.

3. Архитектура СОВ на базе федеративного обучения

В работе предлагается следующая архитектура СОВ, построенная с применением ФО. Основными элементами СОВ являются: компоненты сбора данных или сенсоры безопасности; компоненты анализа данных, которые реализуют различные стратегии обнаружения атак и аномалий; хранилища данных, в которых содержатся как исходные «сырые» события безопасности, так и результаты анализа; база знаний для механизмов обнаружения вторжений и аномалий. В состав СОВ также часто включают компонент оценки рисков и принятия контрмер, на вход которого подается информация о конфигурации защищаемой системы. База знаний СОВ должна постоянно обновляться для актуализации перечня детектируемых информационных угроз. В случае «облачных» СОВ потоки данных от сенсоров безопасности также направляются в облачный центр безопасности, что позволяет проводить глобальную аналитику безопасности, а также разрабатывать новые методы обнаружения вторжений на их основе [32].

Применение ФО для построения СОВ позволяет локально настраивать механизмы обнаружения на основе МО с учетом данных, собираемых локально, и регулярно их обновлять без передачи данных на облачный сервер безопасности. Таким образом, использование ФО изменяет существующие потоки данных в среде СОВ. Структура облачного сервера СОВ расширяется за счет компонентов, отвечающих за организацию и координацию ФО, а структура агента СОВ дополняется компонентами, отвечающими за работу с локальной моделью.

На рис. 3 представлена структурная схема интеллектуальной СОВ (в виде внешних и внутренних информационных объектов, выделенных подсистем и решаемых ими задач), которая использует ФО для настройки и обновления компонентов анализа данных (использованы следующие обозначения: синие прямоугольники — подсистемы СОВ, зеленые прямоугольники — решаемые задачи; желтые объекты — внутренние информационные объекты: шестиугольник — аналитическая модель, цилиндр — репозиторий данных; пунктирные фигуры — дублируемые элементы; облако — информационные объекты или сети).

Схема отражает следующие элементы и логические связи между ними:

1) сеть — внешняя система устройств с каналами передачи данных, подверженная атакам и защищаемая СОВ; сети принадлежат разным организациям;

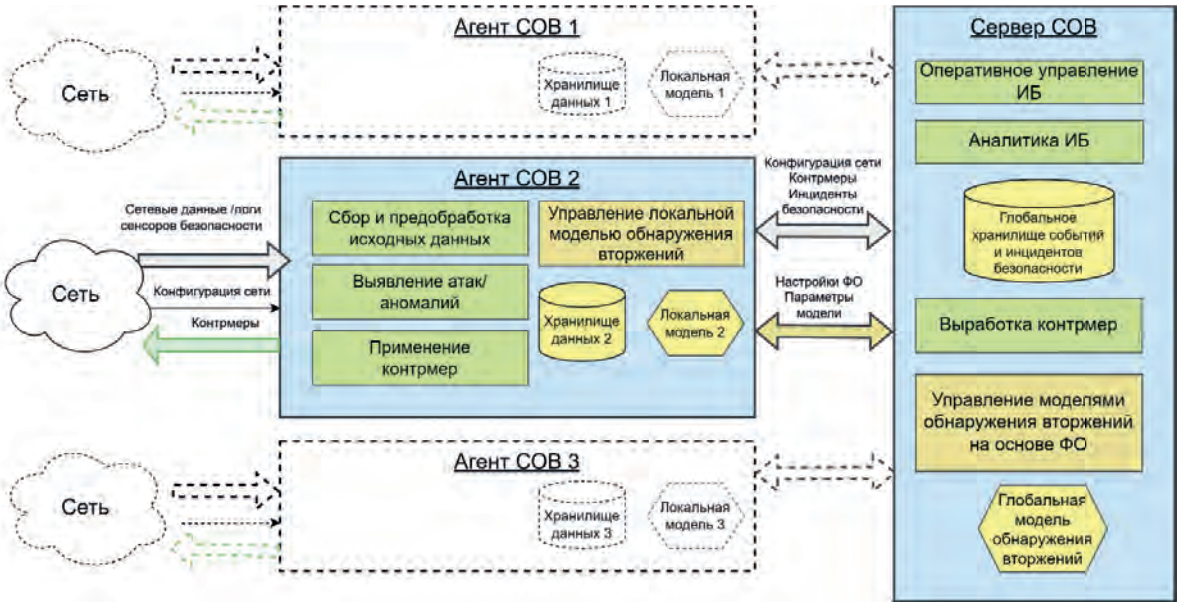

Рис. 3. Структурная схема СОВ, использующая ФО

2) агент СОВ — интеллектуальный агент распределенной СОВ, который обеспечивает анализ состояния и процессов контролируемой сети, выполняет выявление атак и противодействие им;

3) сбор и предобработка исходных данных — задача сбора событий безопасности от различных устройств и приложений, установленных в сети, а также приведения их к виду, подходящему для дальнейшего анализа (включая нормализацию событий, их агрегирование и фильтрацию);

4) выявление атак/аномалий — задача выявления атак и/или аномалий различными методами, реализованными в СОВ; в данной работе рассматриваются методы на базе МО;

5) применение контрмер — задача применения мер противодействия выявленным атакам с учетом текущей конфигурации контролируемой сети; в настоящей работе считается, что выработка контрмер осуществляется сервером СОВ;

6) обучение локальной модели обнаружения вторжений — задача обучения и обновления модели обнаружения вторжений в режиме ФО;

7) локальная модель — аналитическая модель, построенная на основе локальных данных, но с учетом параметров глобальной модели, и выполняющая задачу обнаружения вторжений;

8) хранилище данных — хранилище локальных данных от сенсоров безопасности, которые используются для обучения и обновления модели обнаружения вторжений;

9) сервер СОВ — центральная функциональная подсистема, обеспечивающая координирование всех агентов СОВ, собирающая информацию от них об инцидентах безопасности для формирования глобальной аналитики, настраивающая аналитические модели обнаружения вторжений и вырабатывающая контрмеры на основе полученной информации о конфигурации защищаемой сети и выявленных инцидентах безопасности;

10) оперативное управление информационной безопасностью (ИБ) — задача формирования ситуационной осведомленности о выявленных инцидентах для реагирования на них;

11) аналитика ИБ — задача формирования аналитических отчетов об инцидентах безопасности, выявления основных трендов при нарушении безопасности и т. д.;

12) глобальное хранилище событий и инцидентов безопасности — единая база для всех выявленных инцидентов, служащая для формирования различных механизмов обнаружения вторжений, а также для оценки эффективности глобальной(-ых) модели (моделей) обнаружения вторжений, обучаемых в федеративном режиме.

13) выработка контрмер — задача создания и передачи контрмер конкретным агентам на основании обнаруженных атак, текущей конфигурации сети и процедур противодействия;

14) управление моделями обнаружения вторжений на основе ФО — задача координации процесса об-

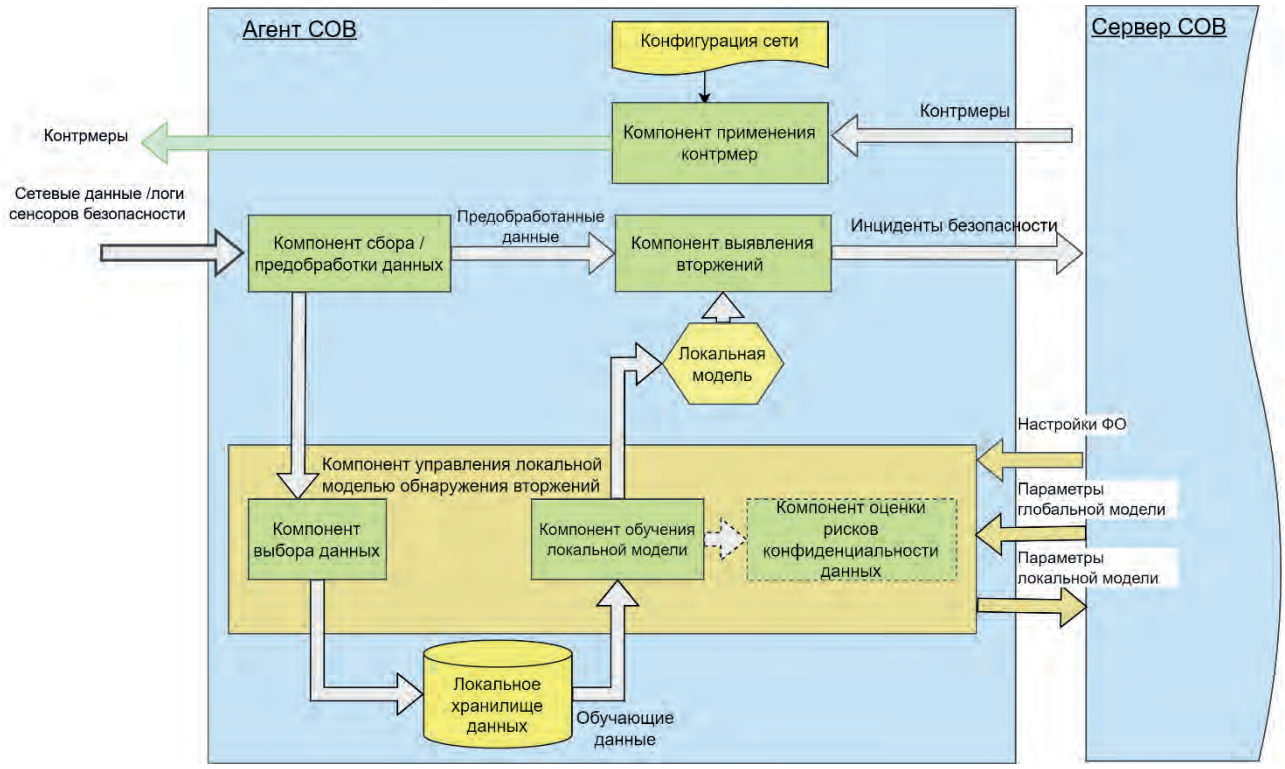

Рис. 4. Структурная схема агента СОВ

учения модели в федеративном режиме, включая выбор агентов СОВ, участвующих в обучении, настройку их конфигурации, обмен параметрами моделей и т. д.;

15) глобальная модель обнаружения вторжений — аналитическая модель, полученная путем синхронизации всех локальных моделей агентов; для простоты будем считать, что такая модель одна, хотя очевидно, что их может быть множество для разных задач и источников данных.

Согласно схеме (см. рис. 3), сервер СОВ выполняет роль агрегирующего сервера ФО, управляя и координируя действия множества обособленных агентов СОВ, отвечающих за безопасность некоторой компьютерной сети(ей). Каждый агент СОВ настраивает и до-обучает глобальную модель на собственном наборе данных, а также выполняет обнаружение атак с ее помощью. При этом, полученные локальные аналитические модели агентов объединяются в общую глобальную модель на сервере, расширяя тем самым диапазон детектируемых ею атак. Следует отметить, что при этом сервер СОВ, как и раньше выполняет роль центра мониторинга ИБ, собирая и агрегируя информацию о выявленных инцидентах безопасности. Полученная таким образом информация может быть использована, в том числе, для валидации и оценки эффективности глобальной модели. Как и в классических решениях по предупреждению вторжений, задача выбора контрмер решается сервером СОВ.

Рассмотрим далее более детально функциональность каждого компонента СОВ — сервера и интеллектуального агента. На рис. 4 представлена функциональная схема агента СОВ (в виде ее функциональных компонентов и обрабатываемых информационных объектов).

Агент СОВ включает следующие элементы:

1) компонент сбора и предобработки данных — осуществляет сбор сетевых данных и логов сенсоров безопасности, а также их обработку для дальнейшего использования;

2) компонент управления локальной модели обнаружения вторжений — выполняет основные функции по формированию обучающей выборки и обучению модели анализа в федеративном режиме в соответствии с полученными настройками ФО;

3) локальное хранилище данных — используется для хранения предобработанных данных, необходимых для обучения локальной модели;

4) компонент выявления вторжений — обнаруживает инциденты безопасности (используя для этого обученную локальную модель обнаружения вторжений) и передает их серверу СОВ;

5) компонент применения контрмер — получает от сервера СОВ контрмеры для противодействия атакам и выполняет их;

6) локальная модель — аналитическая модель анализа, используемая агентом для выявления вторжений.

Компонент управления локальной моделью обнаружения вторжений состоит из трех следующих компонентов: (1) выбора данных, (2) обучения локальной модели и (3) оценки рисков конфиденциальности данных. Компонент выбора данных отвечает за отбор и разметку новых данных, необходимых для настройки локальной модели. Наличие данного модуля связано, в первую очередь, с ограниченными возможностями интеллектуального агента СОВ по хранению данных; например, некоторые сетевые роутеры, которые используются для развертывания системы «умного» дома, имеют только 32Mb памяти для хранения данных. Другой причиной ввода данного компонента в систему является непрерывность генерации данных, в результате чего необходимо в режиме реального времени принимать решение о том, сохранять их или нет. Для реализации компонента может быть использован подход, предложенный в [33], который заключается в оценке поступающих данных в контексте параметров локальной модели и отборе тех образцов, которые соответствуют распределению локальных и глобальных данных, что в конечном итоге уменьшает уровень их разнородности между клиентами и позволяет повысить эффективность ФО. Компонент обучения локальной модели отвечает непосредственно за выполнение ФО, настройки для его выполнения передаются сервером СОВ. Компонент получает параметры глобальной модели, инициализирует с их помощью локальную модель и обновляет ее с учетом данных, накапливаемых локально. Компонент оценки рисков конфиденциальности данных, используемых для выполнения, является необязательным, поскольку его задачей является отслеживание и расчет рисков утечек конфиденциальной информации непосредственно во время обучения. Для реализации компонента может быть использован подход, предложенный в [34], согласно которому оценка рисков учитывает как уровень критичности различных атрибутов, используемых при обучении модели, так и взаимную информационную связь между ними и параметрами модели, передаваемыми серверу СОВ.

На рис. 5 представлена функциональная схема сервера СОВ (в виде ее функциональных компонентов и обрабатываемых информационных объектов). Сервер СОВ включает следующие элементы:

1) компонент выработки контрмер — формирует контрмеры на основе текущей конфигурации защищаемой сети и наблюдаемых инцидентов безопасности;

2) компонент аналитики ИБ — выполняет функцию формирования различных отчетов на основе инцидентов безопасности, выявленных в различных сетях;

3) компонент оперативного управления ИБ — реализует текущий мониторинг различных инцидентов безопасности, выявляемых в различных организациях;

4) компонент управления моделями обнаружения вторжений на основе ФО, состоящий из трех основных компонентов - выбора настроек ФО, агрегации данных и обнаружения атак на ФО;

5)глобальное хранилище событий и инцидентов безопасности — используется для хранения полного набора событий и инцидентов безопасности от всех агентов в интересах последующего централизованного анализа;

6) глобальная модель обнаружения вторжений — в общем случае представляет собой репозитории различных моделей обнаружения вторжений.

Компонент управления моделями обнаружения вторжений на основе ФО состоит из трех следующих компонентов: выбора настроек ФО, агрегации данных и обнаружения атак на ФО. Компонент выбора настроек ФО отвечает за общие настройки агентов СОВ, такие как схема анализируемых атрибутов, тип аналитической модели, ее архитектура, число раундов формирования глобальной модели, число локальных эпох обучения и т.д. Данный компонент на основе анализа конфигурации агента СОВ (а именно, в зависимости от доступных вычислительных ресурсов, объема локального хранилища, уровня доверия защищаемой сети) также определяет стратегию выбора клиентов в процессе обучения, функцию агрегирования данных и т.д. Таким образом, компонент как настраивает параметры ФО на сервере СОВ, так и передает параметры для выполнения локального обучения на агентах СОВ. Компонент агрегации данных выполняет вычисление глобальной модели на основе параметров локальной. На этапе инициализации для этого может быть использована модель, предобученная на некотором доступном наборе данных с похожими характеристиками: общими атрибутами, близкими вероятностями распределения их значений и т.д. Компонент обнаружения атак на ФО предназначен для выявления атак, направленных непосредственно на ФО и глобальную модель анализа. Поскольку ФО построено на взаимодействии некоторого множества агентов, то их присутствие рас-

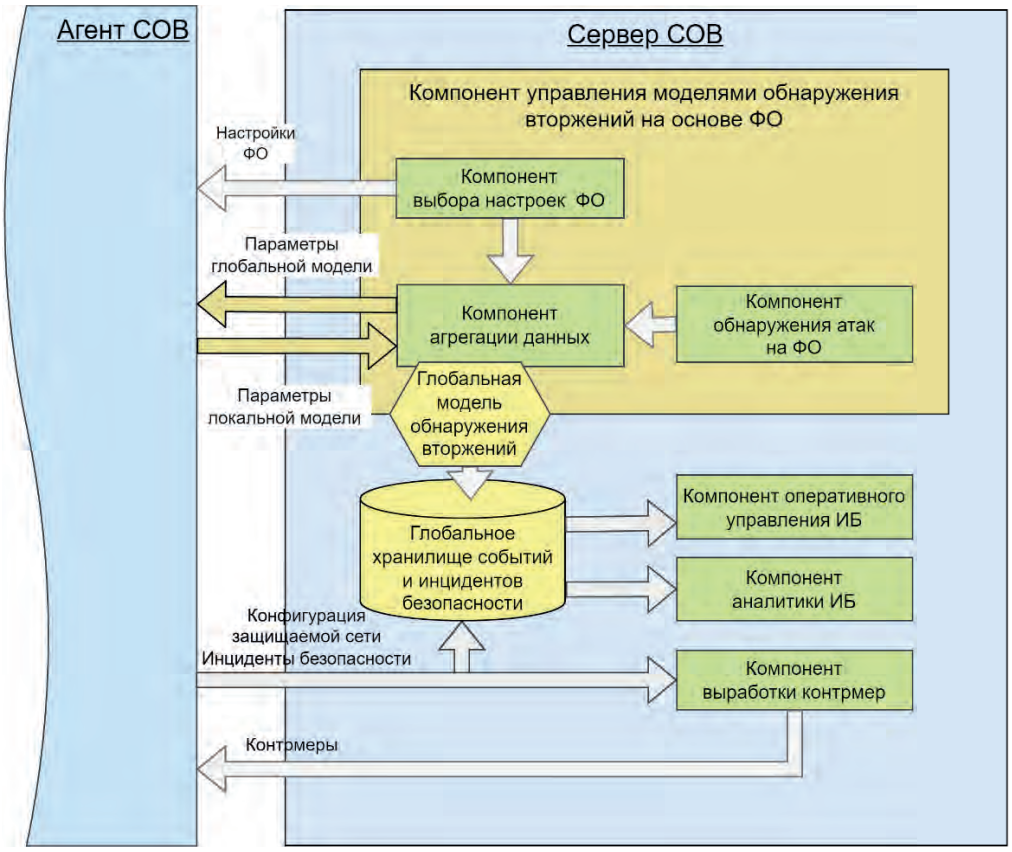

Рис. 5. Структурная схема сервера СОВ

ширяет поверхность вектора атак и актуализирует вопросы, связанные, в первую очередь, с защитой целостности и аутентичности данных, используемых при обучении. Атаки могут быть направлены на модификацию как данных, так и параметров передаваемой модели. Для выявления и противодействия таким атакам предложены различные подходы, в основе которых могут лежать следующие принципы: оценка расстояния между передаваемыми параметрами локальных моделей [35], исключение *k* самых больших и самых маленьких значений параметров модели [36], использование дополнительного тестового набора данных [37] и др.

4. Экспериментальная оценка

Для выполнения экспериментальной оценки компонента СОВ на основе ФО были решены следующие задачи: (1) построение экспериментального стенда; (2) моделирование разбиения данных между агентами СОВ; (3) определение метрик, позволяющих оценить как эффективность модели, обученной в федеративном режиме, так и определить требования к ресурсам агента СОВ для выполнения данной задачи.

Для организации федеративного обучения между агентами был выбран программный проект Flower [38]. Данный проект не требователен к вычислительным ресурсам и легко настраивается. Взаимодействие между клиентами и сервером ФО осуществляется на основе технологии gRPC. Проект предусматривает два режима запуска ФО — симуляционный и федеративный. В симуляционном режиме возможно запустить систему ФО (один сервер и несколько клиентов) на одной вычислительной машине. Имеется также возможность предварительной оценки производительности ФО на основе средств мониторинга системных ресурсов, обеспечивающих определение загрузки ЦПУ, потребления оперативной памяти и объема передаваемого сетевого трафика. В федеративном режиме система разворачивается на реальных физических устройствах. Кроме того, фреймворк проекта имеет хорошо описанные программные интерфейсы для подключения собственных алгоритмов агрегирования локальных моделей и на текущий момент поддерживает различные алгоритмы агрегирования, устойчивые к неидентично распределенным данным [39,40]. Например, в него включены недавно предложенные стратегии для гетерогенных и неидентичных данных:

— распределенный алгоритм агрегации на основе расчета медианных значений для параметров модели, который использует только один раунд связи между клиентами, что позволяет более высокую эффективность передаче данных [41];
— адаптивная стратегия оптимизации на стороне сервера, учитывающая гетерогенность устройств [39];
— FedBN-стратегия оптимизации, выполняемая на стороне клиента, которая использует локальные слои пакетной нормализации для решения проблем сдвига в неидентично распределенных данных[40].

В литературе предложено два следующих варианта моделирования распределения данных между клиентами [42]:

— один набор данных распределяется между *N* клиентами так, чтобы каждый клиент отвечал за определенный тип атак;
— каждый клиент получает свой набор данных, у которых одинаковое количество атрибутов.

В настоящей работе был использован первый вариант моделирования распределения данных между клиентами. В качестве основного набора данных использовался CIC-IDS2017 [22]. Он содержит информацию о сетевых потоках за 5 дней функционирования небольшой компьютерной сети, состоящей из 10 рабочих станций, в виде описательных статистик различных параметров (например, длины пакетов). Логи первого дня соответствуют нормальному функционированию системы, а данные за остальные четыре дня содержат информацию о различных сетевых атаках. Для выполнения экспериментов набор CIC-IDS2017 был поделен на две части таким образом, чтобы в состав каждого набора входили данные о разных типах атак. В табл. 1 приведена структура обоих наборов данных, которые использовались в экспериментах; во всех экспериментах набор данных был разделен на обучающую и тестовую выборки в соотношении 80 к 20.

С учетом того, что в каждом наборе данных представлены разные типы атак, было принято решение сформулировать задачу обнаружения вторжений в виде задачи бинарной классификации — т.е. определения того, является ли сетевой поток нормальным или нет.

Структура сверточной нейронной сети, которая во всех экспериментах обучалась 5 эпох, представлена на рис. 6.

Таблица 1

Структура наборов данных, используемых в экспериментах

Класс	Число сетевых потоков
Набор данных (DS1)	
Норма	251547
Атака Brute Force FTP	7916
Атака Brute Force SSH	4928
Dos-атака GoldenEye	15146
Dos-атака Hulk	28216
Dos-атака Slowhttptest	5621
Dos-атака Slowris	6081
Атака Hearbleed port 444	2
Набор данных (DS2)	
Норма	251547
Атака на проникновение Cool disk (MAC OS)	2
Атака на проникновение Dropbox	10
Веб-атака с помощью sql инъекций	18
Веб-атака типа межсайтовый скриптинг (XSS-атака)	1324
Веб-атака методом прямого перебора	2700
Ботсеть ARES - sql	1470
DDoS -атака LOIT	90418
Сканирования портов (с межсетевым экраном)	728
Сканирования портов (без межсетевым экраном)	318756

Поскольку процесс ФО должен выполняться регулярно во время функционирования СОВ, то возникает необходимость оценить его влияние на функционирование самого агента; в данной работе в качестве такого параметра было выбрано время обучения и оценка этого параметра в зависимости от настроек ФО. Как результат, для оценки эффективности были использованы следующие метрики: точность глобальной модели (accuracy, precision and recall и F-мера), время обучения, потребление памяти и загрузка процессора.

Для оценки целесообразности применения ФО при построении аналитических моделей обнаружения

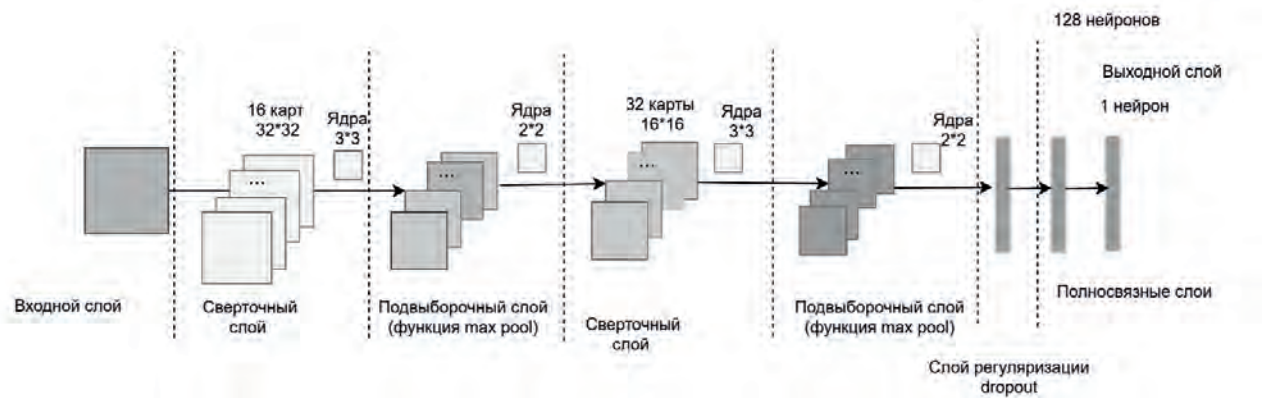

Рис. 6. Структура сверточной нейронной сети в экспериментах

Таблица 2

Точность обнаружения атак моделей, обученных только на локальных наборах данных

Обучающая выборка	Метрика	Тестовый набор данных	
		Набор данных DS1	Набор данных DS2
Набор данных DS1	Accuracy	0.99	0.48
	Recall	0.99	0.99
	Precision	0.99	0.42
	F-measure	0.99	0.59
Набор данных DS2	Accuracy	0.85	0.99
	Recall	0.99	0.99
	Precision	0.84	0.99
	F-measure	0.91	0.99

вторжений был разработан следующий сценарий эксперимента.

На первом этапе обучение модели осуществлялось на одном из наборов данных, а затем выполнялась оценка ее эффективности на втором наборе данных. Такая схема позволяет смоделировать ситуацию, когда некоторая обученная модель обнаружения «сталкивается» с новыми, ранее неизвестными ей типами атак, а полученные результаты позволяют судить об ее обобщающей способности. В этой серии экспериментов нейронная сеть обучалась 5 эпох. Полученные результаты представлены в табл. 2.

Очевидно, что точность детектирования сильно зависит от того, какие типы атак были представлены в обучающей выборке, и модель, обученная на наборе данных DS2 (который содержит более разнообразный набор атак), обнаруживает новые атаки более точно. Модель, которая обучалась на наборе данных DS1 (который содержит в основном DoS-атаки), обнаруживает атаки значительно хуже — ее точность составляет 48%, а низкое значение метрики precision (42%) говорит о высоком уровне ложноположительных срабатываний.

На втором этапе выполнялось обучение в федеративном режиме, моделировалось взаимодействие двух клиентов со следующими ресурсами: ЦПУ Intel Core i5-8 265U с тактовой частотой 1.80 ГГц и объемом оперативной памяти 8 ГБ. На одном клиенте в качестве обучающей выборки был использован набор данных DS1, а на втором - набор данных DS2.

Модель также обучалась 5 эпох, агрегация параметров локальных моделей выполнялась в следующих режимах: (1) каждый раунд; (2) каждые два раунда и (3) один раз в конце обучения. В качестве функции агрегации была использована функция FederatedAveraging. Результаты эксперимента приведены в табл. 3. Из нее следует, что точность модели, обученной в федеративном режиме, достаточно высокая на обоих наборах данных, благодаря объединению параметров локальных моделей, обученных

Таблица 3

Результаты обучения нейронной сети в федеративном режиме

Число раундов агрегации	Время обучения	Метрика	Тестовый набор данных	
			Набор данных DS1	Набор данных DS2
5 (после каждой эпохи обучения)	71.5 мин	Accuracy	0.94	0.99
		Recall	0.99	0.99
		Precision	0.93	0.99
		F-measure	0.96	0.99
3 (через каждые 2 эпохи)	80.0 мин	Accuracy	0.90	0.98
		Recall	0.99	0.99
		Precision	0.88	0.95
		F-measure	0.94	0.97
1 (в конце обучения)	73.1 мин	Accuracy	0.85	0.99
		Recall	0.99	0.99
		Precision	0.84	0.98
		F-measure	0.91	0.99

на каждом их них по отдельности. Следует отметить, что чем чаще происходило объединение моделей, т.е. чем выше было число раундов агрегации, тем выше точность модели была на обоих наборах данных. Длительность ФО в табл. 3 указана для агрегирующего сервера. В данном эксперименте зависимости длительности обучения от числа раундов агрегирования выявлено не было. Следует также отметить, что время локального обучения для двух клиентов было разным, для клиента с набором DS2 оно всегда было в 1.6 больше, чем с DS1, что закономерно, так как число записей в этом наборе в два раза больше числа записей в наборе DS1.

Средняя загрузка ЦПУ для обоих клиентов составила 40%, загрузка оперативной памяти составила 300-400МБ во время обучения.

Таким образом, можно сделать выводы, что, как и в случае классического машинного обучения, требования к ресурсам компонентов СОВ на основе ФО будут определяться объемом данных, которые используются для обучения, а также сложностью обучаемой модели. Соответственно, для систем с ограниченными вычислительными ресурсами возможно использование только легковесных и простых моделей МО. Повышение сложности аналитических моделей повлечет за собой повышение требований к вычислительным ресурсам, оперативной памяти и объему жесткого диска. Тем не менее целесообразность использования федеративного обучения для построения глобальных моделей обнаружения вторжений очевидна, оно естественным образом позволяет расширить множество детектируемых атак за счет увеличения обучающей выборки.

5. Заключение

В настоящее время в научном сообществе исследуются возможности федеративного обучения как механизма обмена знаниями об угрозах и атаках без обмена реальными данными. Предложенные решения, так и проведенные в данной работе эксперименты показывают эффективность его использования при построении моделей МО, предназначенных для выявления вторжений. Однако применение ФО в задачах обнаружения вторжений связано с решением многих практических задач, в частности, каким образом должны быть построены такие системы, как выполнять оценку их эффективности, как следует учитывать влияние ФО на функционирование всего компонента СОВ в целом, составным компонентом которого оно является.

В настоящей работе рассмотрена общая концепция ФО, предложена архитектура СОВ на основе ФО, представлены основные компоненты СОВ с описанием их функциональности. Также в работе приведены результаты экспериментальной оценки компонента СОВ, отвечающего за обнаружение вторжений. Дан-

ные оценки включают не только метрики точности выявления атак, но и характеристики потребления ресурсов на выполнение ФО.

В настоящий момент оценка влияния ФО ограничивается только оценкой длительности обучения, загрузкой ЦПУ и потреблением оперативной памяти, однако, она должна также включать данные по сетевому трафику для получения полноценного представления о необходимых и достаточных характеристиках компонента СОВ, отвечающего за локальное обучение на основе ФО. В связи с этом будущие работы включают в себя разработку системы мониторинга ФО, выполнения экспериментов с различными настройками ФО, включая оценку различных стратегий агрегирования. Отдельно следует выделить задачу по определению методики отбора данных для выполнения локального обучения, в том числе определения «эффективного» размера обучающей выборки.

Благодарность. *Исследование выполнено за счет гранта Российского научного фонда № 22-21-00724, https://rscf.ru/project/22-21-00724/.*

Рецензент: *Лаута Олег Сергеевич, доктор технических наук, профессор кафедры комплексного обеспечения информационной безопасности Государственного университета морского и речного флота имени адмирала С.О. Макарова, Санкт-Петербург, Россия.*

E-mail: laos-82@yandex.ru

Литература

1. Kotenko I., Izrailov K., Buinevich M. Static Analysis of Information Systems for IoT Cyber Security: A Survey of Machine Learning Approaches // Sensors. 2022. Vol. 22. Iss. 4. 1335. DOI: 10.3390/s22041335.
2. Израилов К.Е., Буйневич М.В. Метод обнаружения атак различного генеза на сложные объекты на основе информации состояния. Часть 1. Предпосылки и схема // Вопросы кибербезопасности. 2023. № 3(55). С. 90-100. DOI: 10.21681/2311-3456-2023-3-90-100.
3. Израилов К.Е., Буйневич М.В. Метод обнаружения атак различного генеза на сложные объекты на основе информации состояния. Часть 2. Алгоритм, модель и эксперимент // Вопросы кибербезопасности. 2023. № 4(56). С. 80-93. DOI: 10.21681/2311-3456-2023-4-80-93.
4. Котенко И.В., Саенко И.Б., Лаута О.С., Крибель А.М. Метод раннего обнаружения кибератак на основе интеграции фрактального анализа и статистических методов // Первая миля. 2021. № 6 (98). С. 64-71. DOI: 10.22184/2070-8963.2021.98.6.64.70.
5. Котенко В.И., Саенко И.Б., Коцыняк М.А., Лаута О.С. Оценка киберустойчивости компьютерных сетей на основе моделирования кибератак методом преобразования стохастических сетей // Труды СПИИРАН. 2017. № 6(55). С. 160-184. DOI: 10.15622/sp.55.7.
6. Branitskiy A., Kotenko I., Saenko I. Applying Machine Learning and Parallel Data Processing for Attack Detection in IoT // IEEE Transactions on Emerging Topics in Computing, 2021, vol. 9, no. 4. P. 1642-1653. DOI: 10.1109/TETC.2020.3006351.
7. Tushkanova O., Levshun D., Branitskiy A., Fedorchenko E., Novikova E., Kotenko I. Detection of Cyberattacks and Anomalies in Cyber-Physical Systems: Approaches, Data Sources, Evaluation // Algorithms. 2023. 16(2):85. DOI: 10.3390/a16020085.
8. McMahan H. B., Moore E., Ramage D., Hampson S., Arcas B.A.Y. Communication-efficient learning of deep networks from decentralized data // International Conference on Artificial Intelligence and Statistics, 2016. URL: https://api.semanticscholar.org/CorpusID:14955348 (дата обращения: 20.08.2023).
9. Романов Н.Е., Израилов К.Е., Покусов В.В. Система поддержки интеллектуального программирования: машинное обучение feat. быстрая разработка безопасных программ // Информатизация и связь. 2021. № 5. С. 7-17. DOI: 10.34219/2078-8320-2021-12-5-7-16.
10. Astillo P.V., Duguma D.G., Park H., Kim J., Kim B., and You I.. Federated intelligence of anomaly detection agent in IoTmd-enabled diabetes management control system // Future Generation Computer Systems, 128. 2022. P.395–405. ISSN 0167-739X. DOI: 10.1016/j.future.2021.10.023.
11. Campos E.M., Saura P.F., Gonzalez-Vidal A., Hernandez-Ramos J., Bernabe J., Baldini G., and Skarmeta A. Evaluating federated learning for intrusion detection in internet of things: Review and challenges // Computer Networks, 2022. 203:108661. ISSN 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2021.108661.
12. Fedorchenko E., Novikova E., and Shulepov A. Comparative review of the intrusion detection systems based on federated learning: Advantages and open challenges // Algorithms, 15(7), 2022. ISSN 1999-4893. DOI: 10.3390/a15070247.
13. Friha O., Ferrag M. A., Shu L., Maglaras L., Choo K.-K., and Nafaa M. Felids: Federated learning-based intrusion detection system for agricultural internet of things // Journal of Parallel and Distributed Computing, 165, 2022. P.17–31. ISSN 0743-7315. DOI: 10.1016/j.jpdc.2022.03.003.
14. Bonawitz K., Ivanov V., Kreuter B., Marcedone A., McMahan H.B., Patel S., Ramage D., Segal A., and Seth K. Practical secure aggregation for privacy-preserving machine learning // Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, New York, NY, USA, 2017. Association for Computing Machinery. P.1175–1191. ISBN 9781450349468. DOI: 10.1145/3133956.3133982.
15. Stevens T., Skalka C., Vincent C., Ring J., Clark S., and Near J. Efficient differentially private secure aggregation for federated learning via hardness of learning with errors // 31st USENIX Security Symposium (USENIX Security 22). Boston, MA, August 2022. USENIX

Association. P.1379–1395. ISBN 978-1-939133-31-1. URL:https://www.usenix.org/conference/usenixsecurity22/ presentation/ stevens (дата обращения: 20.08.2023).
16. Aouedi O., Piamrat K., Muller G., and Singh K. Fluids: Federated learning with semi- supervised approach for intrusion detection system // 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC). 2022. P.523–524. DOI: 10.1109/ CCNC49033.2022.9700632.
17. Qin Y. and Kondo M. Federated learning-based network intrusion detection with a feature selection approach. // 2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE). 2021. P.1–6. DOI: 10.1109/ ICECCE52056.2021.9514222.
18. Fan Y., Li Y., Zhan M., Cui H., and Zhang Y. Iotdefender: A federated transfer learning intrusion detection framework for 5G IoT // 2020 IEEE 14th International Conference on Big Data Science and Engineering (BigDataSE). 2020. P.88–95. DOI:10.1109/ BigDataSE50710.2020.00020.
19. Nguyen T.D., Marchal S., Miettinen M., Fereidooni H., Asokan N., and Sadeghi A.-R. Diot: A federated self-learning anomaly detection system for IoT // Proc. 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). 2019. P.756–767.
20. Rey V., Sanchez P.M.S., Celdran A.H., and Bovet G. Federated learning for malware detection in IoT devices // Computer Networks, 204:108693, 2022. ISSN 1389-1286. DOI: 10.1016/j.comnet.2021.108693.
21. Meidan Y., Bohadana M., Mathov Y., Mirsky Y., Shabtai A., Breitenbacher D., and Elovici Y. N-baiot—network-based detection of IoT botnet attacks using deep autoencoders // IEEE Pervasive Computing, 17(3). 2018. P.2–22, DOI: 10.1109/MPRV.2018.03367731.
22. Sharafaldin I., Lashkari A.H., and Ghorbani A. Toward generating a new intrusion detection dataset and intrusion traffic characterization // Proc. of 4th International Conference on Information Systems Security and Privacy. 2018. P.108–116. DOI: 10.5220/0006639801080116.
23. Vaccari I., Chiola G., Aiello M., Mongelli M., Cambiaso E. MQTTset, a New Dataset for Machine Learning Techniques on MQTT // Sensors. 2020; 20(22):6578. https://doi.org/10.3390/s20226578.
24. Elsayed M. S., Le-Khac N.-A. and Jurcut A. D. InSDN: A Novel SDN Intrusion Dataset // IEEE Access, vol. 8. 2020. P.165263-165284. DOI: 10.1109/ACCESS.2020.3022633.
25. Rodrıguez-Barroso N., Stipcich G., Jimenez-Lopez D., Ruiz-Millan J.A., Martınez-Camara E., Gonzalez-Seco G., M. Luzon V., Veganzones M., and Herrera F. Federated learning and differential privacy: Software tools analysis, the sherpa.ai fl framework and methodological guidelines for preserving data privacy // Information Fusion, 64. 2020. P.270–292. ISSN 1566-2535. DOI: 10.1016/j. inffus.2020.07.009.
26. Sarhan M., Lo W.W., Layeghy S., and Portmann M. Hbfl: A hierarchical blockchain-based federated learning framework for a collaborative IoT intrusion detection, 2022. URL:https://arxiv.org/ abs/2204.04254 (дата обращения: 20.08.2023).
27. Moustafa N. The BoT-IoT dataset, 2019. URL https://dx.doi.org/10.21227/r7v2-x988 (дата обращения: 20.08.2023).
28. Abdel-Basset M., Moustafa N., Hawash H., Razzak I., Sallam K., and Elkomy O. Federated intrusion detection in blockchain-based smart transportation systems // IEEE Transactions on Intelligent Transportation Systems, 23(3). 2022. P.2523–2537. DOI: 10.1109/ TITS.2021.3119968.
29. Liu H., Zhang S., Zhang P., Zhou X., Shao X., Pu G., and Zhang Y. Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing // IEEE Transactions on Vehicular Technology, 70(6), 2021. P.6073–6084. DOI: 10.1109/ TVT.2021.3076780.
30. Chai H., Leng S., Chen Y., and Zhang K. A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles // IEEE Transactions on Intelligent Transportation Systems, 22(7), 2021. P.3975–3986. ISSN 1524-9050. DOI:10.1109/ TITS.2020.3002712.
31. Novikova E., Doynikova E., and Golubev S. Federated learning for intrusion detection in the critical infrastructures: Vertically partitioned data use case // Algorithms, 15(4), 2022. ISSN 1999-4893. doi:10.3390/a15040104.
32. Saputra F.A., Salman M., Hasim J.N., Nadhori I.U., Ramli K. The next-generation NIDS platform: Cloud-based snort NIDS using containers and big data // Big Data and Cognitive Computing, 6(1), 2022. ISSN 2504-2289. doi:10.3390/bdcc6010019.
33. Gong C., Zheng Z., Wu F., Shao Y., Li B., and Chen G.. To store or not? online data selection for federated learning with limited storage // Proc. of the ACM Web Conference 2023, WWW '23. New York, NY, USA, 2023. Association for Computing Machinery. P.3044–3055. ISBN 9781450394161. DOI: 10.1145/3543507.3583426.
34. Jiang C., Xia C., Liu Z., and Wang T. Feddroidmeter: A privacy risk evaluator for fl- based android malware classification systems. Entropy, 25(7), 2023. ISSN 1099-4300. DOI: 10.3390/e25071053.
35. Blanchard P., El Mhamdi E.M., Guerraoui R., and Stainer J. Machine learning with adversaries: Byzantine tolerant gradient descent // Proc. of the 31st International Conference on Neural Information Processing Systems, NIPS'17. Red Hook, NY, USA. Curran Associates Inc. 2017. P.118–128. ISBN 9781510860964.
36. Yin D., Chen Y., Kannan R., and Bartlett P. Byzantine-robust distributed learning: Towards optimal statistical rates // Proc. of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research. PMLR, 10–15 Jul 2018. P.5650–5659.
37. Cao X., Fang M., Liu J., and Gong N.J. Fltrust: Byzantine-robust federated learning via trust bootstrapping. CoRR, abs/2012.13995, 2020. URL: https://arxiv.org/abs/2012.13995. (дата обращения: 20.08.2023).
38. Flower — фреймворк для федеративного обучения. URL https://flower.dev/. (дата обращения: 20.08.2023).
39. Li X., Jiang M., Zhang X., Kamp M., and Dou Q. Fedbn: Federated learning on non-iid features via local batch normalization. CoRR, abs/2102.07623, 2021. (дата обращения: 20.08.2023).
40. Reddi S.J., Charles Z., Zaheer M., Garrett Z. , Rush K., Konecny J., Kumar S., and McMahan H.B. Adaptive federated optimization. CoRR, abs/2003.00295, 2020. (дата обращения: 20.08.2023).
41. Yin D., Chen Y., Ramchandran K., Bartlett P.L. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. International Conference on Machine Learning. 2018. URL: https://api.semanticscholar.org/CorpusID:3708326 (дата обращения: 20.08.2023).
42. Новикова Е.С., Федорченко Е.В., Котенко И.В., Холод И.И. Аналитический обзор подходов к обнаружению вторжений, основанных на федеративном обучении: преимущества использования и открытые задачи // Информатика и автоматизация, 22(5). С.1034–1082. DOI:10.15622/ia.22.5.4.

FEDERATED LEARNING BASED INTRUSION DETECTION: SYSTEM ARCHITECTURE AND EXPERIMENTS

Novikova E.S.[5], Kotenko I.V.[6], Meleshko A.V.7, Izrailov K.E.[8]

The goal of the investigation: *to develop an approach to building an intrusion detection system based on federated machine learning.*

Result: *the concept and architecture of an intrusion detection system based on federated machine learning is developed. The proposed architecture includes new components responsible for the organization of federated learning, such as components of data selection, local model training, sensitive information risk assessment, detection of federated learning attacks, and also defines their links with other functional elements of the system. To perform experimental evaluation of the components of the intrusion detection system based on federated learning, the metrics for evaluating their performance are formulated, they allow one to estimate, among other things, the requirements for the computational resources of the system. An approach to modeling the data distribution between the interacting components is proposed, and experimental evaluations of the intrusion detection performance using machine learning models trained in federated mode are obtained.*

Scientific novelty: *literature analysis has shown that the use of federated learning for building intrusion detection systems is associated with a number of open practical problems; in particular, there is no general methodology for building and evaluating the effectiveness of such systems. This paper proposes an architecture of the intrusion detection system that takes into account the practical features of using federated learning, and also presents the results of experimental evaluation of the effectiveness of intrusion detection models trained in federated mode.*

Contribution: *Novikova E.S. and Kotenko I.V. — the general concept and architecture of an intrusion detection system using federated learning, data collection methodology for researching the security of cyber-physical systems; Novikova E.S. and Izrailov K.E. — development of the functionality of individual components of the intrusion detection system, Meleshko A.V. — performing experiments.*

Keywords: *cybersecurity, cyberphysical systems, detection of anomalies and cyberattacks, distributed machine learning, convolutional neural network, performance assessment.*

References

1. Kotenko I., Izrailov K., Buinevich M. Static Analysis of Information Systems for IoT Cyber Security: A Survey of Machine Learning Approaches . Sensors. 2022. Vol. 22. Iss. 4. pp. 1335. DOI: 10.3390/s22041335.
2. Izrailov K., Buinevich M. [A method for detecting attacks of different genesis on complex objects based on state information. Part 1. Prerequisites and scheme] Метод обнаружения атак различного генеза на сложные объекты на основе информации состояния. Часть 1. Предпосылки и схема. Cybersecurity issues [Вопросы кибербезопасности]. 2023. No 3(55). pp. 90-100. DOI: 10.21681/2311-3456-2023-3-90-100. (in Russian)
3. Izrailov K., Buinevich M. [A method for detecting attacks of different genesis on complex objects based on state information. Part 2. Algorithm, model and experiment] Метод обнаружения атак различного генеза на сложные объекты на основе информации состояния. Часть 2. Алгоритм, модель и эксперимент. Cybersecurity issues [Вопросы кибербезопасности]. 2023. No 4(56). pp. 80-93.

5 Evgenia S. Novikova, Ph.D. (Technology), Associate Professor, Senior Researcher of Laboratory of Computer Security Problems at St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), St. Petersburg, Russia. ORCID: https://orcid.org/0000-0003-2923-4954. Scopus Author ID: 55415626100. E-mail: novikova@comsec.spb.ru.

6 Igor V. Kotenko, Honored Worker of Science of the Russian Federation, Dr.Sc. (Technology), Professor, Principal Researcher and Head of Laboratory of Computer Security Problems of St. Petersburg Federal Research Center of the Russian Academy of Sciences, Saint-Petersburg. ORCID: 0000-0001-6859-7120. Scopus Author ID: 15925268000. E-mail: ivkote@comsec.spb.ru

7 Alexei V. Meleshko, Junior Researcher of Laboratory of Computer Security Problems at St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), St. Petersburg, Russia. ORCID: 0000-0002-1209-4230. Scopus Author ID: 57214672771. E-mail: meleshko.a@iias.spb.su.

8 Konstantin E. Izrailov, Ph.D. (Technology), Associate Professor, Senior Researcher of Laboratory of Computer Security Problems of St. Petersburg Federal Research Center of the Russian Academy of Sciences, Saint-Petersburg. ORCID: 0000-0002-9412-5693. Scopus Author ID: 56123238800. E-mail: konstantin.izrailov@mail.ru.

DOI: 10.21681/2311-3456-2023-4-80-93. (in Russian)
4. Kotenko I., Saenko I., Lauta O., Kribel. [A method for early detection of cyberattacks based on the integration of fractal analysis and statistical methods] Метод раннего обнаружения кибератак на основе интеграции фрактального анализа и статистических методов. Pervaya milya [Первая миля]. 2021. № 6 (98). pp. 64-71. DOI: 10.22184/2070-8963.2021.98.6.64.70
5. Kotenko V.I., Saenko I.B., Kotsynyak M.A., Lauta O.S. [Assessment of Cyber-Resilience of Computer Networks based on Simulation of Cyber Attacks by the Stochastic Networks Conversion Method] Оценка киберустойчивости компьютерных сетей на основе моделирования кибератак методом преобразования стохастических сетей. SPIIRAS Proceedings [Труды СПИИРАН]. 2017. No 6(55). pp.160-184. DOI: https://doi.org/10.15622/sp.55.7.
6. Branitskiy A., Kotenko I., Saenko I. Applying Machine Learning and Parallel Data Processing for Attack Detection in IoT. IEEE Transactions on Emerging Topics in Computing, 2021, vol. 9, no. 4, pp. 1642-1653. DOI: 10.1109/TETC.2020.3006351.
7. Tushkanova O, Levshun D, Branitskiy A, Fedorchenko E, Novikova E, Kotenko I. Detection of Cyberattacks and Anomalies in Cyber-Physical Systems: Approaches, Data Sources, Evaluation. Algorithms. 2023. 16(2):85. DOI: 10.3390/a16020085
8. McMahan H. B., Moore E., Ramage D., Hampson S., Arcas B.A.Y. Communication-efficient learning of deep networks from decentralized data. International Conference on Artificial Intelligence and Statistics, 2016. URL: https://api.semanticscholar.org/CorpusID:14955348 (accessed on: 20.08.2023).
9. Romanov N., Izrailov K., PokusovV. [Intelligent programming support system: machine learning feat. fast development of secure programs] Система поддержки интеллектуального программирования: машинное обучение feat. быстрая разработка безопасных программ. Informatization and communication [Информатизация и связь]. 2021. No 5. pp. 7-17. DOI: 10.34219/2078-8320-2021-12-5-7-16. (in Russian)
10. Astillo P.V., Duguma D.G., Park H., Kim J., Kim B., and You I. Federated intelligence of anomaly detection agent in IoTmd-enabled diabetes management control system. Future Generation Computer Systems, 128:395-405, 2022. ISSN 0167-739X. DOI: 10.1016/j.future.2021.10.023.
11. Campos E.M., Saura P.F., Gonzalez-Vidal A., Hernandez-Ramos J., Bernabe J., Baldini G., and Skarmeta A. Evaluating federated learning for intrusion detection in internet of things: Review and challenges. Computer Networks, 203:108661, 2022. ISSN 1389-1286. doi:https://doi.org/10.1016/j.comnet.2021.108661.
12. Fedorchenko E., Novikova E., and Shulepov A. Comparative review of the intrusion detection systems based on federated learning: Advantages and open challenges. Algorithms, 15(7), 2022. ISSN 1999-4893. DOI::10.3390/a15070247.
13. Friha O., Ferrag M. A., Shu L., Maglaras L., Choo K.-K., and Nafaa M. Felids: Federated learning-based intrusion detection system for agricultural internet of things. Journal of Parallel and Distributed Computing, 165:17–31, 2022. ISSN 0743-7315. DOI: 10.1016/j.jpdc.2022.03.003.
14. Bonawitz K., Ivanov V., Kreuter B., Marcedone A., McMahan H.B., Patel S., Ramage D., Segal A., and Seth K. Practical secure aggregation for privacy-preserving machine learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, pp.1175–1191, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349468. DOI:10.1145/3133956.3133982.
15. Stevens T., Skalka C., Vincent C., Ring J., Clark S., and Near J.. Efficient differentially private secure aggregation for federated learning via hardness of learning with errors. Proc. of 31st USENIX Security Symposium (USENIX Security 22), pp.1379–1395, Boston, MA, August 2022. USENIX Association. ISBN 978-1-939133-31-1. URL:https://www.usenix.org/conference/usenixsecurity22/ presentation/stevens (accessed on: 20.08.2023).
16. Aouedi O., Piamrat K., Muller G., and Singh K. Fluids: Federated learning with semi- supervised approach for intrusion detection system. 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC), pp.523–524, 2022. DOI: 10.1109/CCNC49033.2022.9700632.
17. Qin Y. and Kondo M. Federated learning-based network intrusion detection with a feature selection approach. // 2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), pp.1–6, 2021. DOI: 10.1109/ICECCE52056.2021.9514222.
18. Fan Y., Li Y., Zhan M., Cui H., and Zhang Y. Iotdefender: A federated transfer learning intrusion detection framework for 5G IoT . Proc. of 2020 IEEE 14th International Conference on Big Data Science and Engineering (BigDataSE), pp.88–95, 2020. DOI: 10.1109/BigDataSE50710.2020.00020.
19. Nguyen T.D., Marchal S., Miettinen M., Fereidooni H., Asokan N., and Sadeghi A.-R. Diot: A federated self-learning anomaly detection system for IoT. Proc. 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), pp.756–767, 2019.
20. Rey V., Sanchez P.M.S., Celdran A.H., and Bovet G. Federated learning for malware detection in IoT devices. Computer Networks, 204:108693, 2022. ISSN 1389-1286. DOI: 10.1016/j.comnet.2021.108693.
21. Meidan Y., Bohadana M., Mathov Y., Mirsky Y., Shabtai A., Breitenbacher D., and Elovici Y. N-baiot—network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Computing, 17(3):12–22, 2018. DOI: 10.1109/MPRV.2018.03367731.
22. Sharafaldin I., Lashkari A.H., and Ghorbani A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proc. of 4th International Conference on Information Systems Security and Privacy. pp.108–116, 2018. DOI: 10.5220/0006639801080116.
23. Vaccari I., Chiola G., Aiello M., Mongelli M., Cambiaso E. MQTTset, a New Dataset for Machine Learning Techniques on MQTT . Sensors. 2020; 20(22):6578. https://doi.org/10.3390/s20226578.
24. Elsayed M. S., Le-Khac N. -A. and Jurcut A. D. InSDN: A Novel SDN Intrusion Dataset. IEEE Access, vol. 8, pp. 165263-165284, 2020. DOI: 10.1109/ACCESS.2020.3022633.
25. Rodrıguez-Barroso N., Stipcich G., Jimenez-Lopez D., Ruiz-Millan J.A., Martınez-Camara E., Gonzalez-Seco G., M. Luzon V., Veganzones M., and Herrera F. Federated learning and differential privacy: Software tools analysis, the sherpa.ai fl framework and methodological guidelines for preserving data privacy. Information Fusion, 64:270–292, 2020. ISSN 1566-2535. DOI: 10.1016/j.inffus.2020.07.009.
26. Sarhan M., Lo W.W., Layeghy S., and Portmann M. Hbfl: A hierarchical blockchain-based federated learning framework for a collaborative IoT intrusion detection, 2022. URL:https://arxiv.org/abs/2204.04254 (accessed on: 20.08.2023).
27. Moustafa N. The BoT-IoT dataset, 2019. URL https://dx.doi.org/10.21227/r7v2-x988 (accessed on: 20.08.2023).
28. Abdel-Basset M., Moustafa N., Hawash H., Razzak I., Sallam K., and Elkomy O. Federated intrusion detection in blockchain-based

smart transportation systems. IEEE Transactions on Intelligent Transportation Systems, 23(3):2523–2537, 2022. DOI: 10.1109/TITS.2021.3119968.
29. Liu H., Zhang S., Zhang P., Zhou X., Shao X., Pu G., and Zhang Y. Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing. IEEE Transactions on Vehicular Technology, 70(6):6073–6084, 2021. DOI: 10.1109/TVT.2021.3076780.
30. Chai H., Leng S., Chen Y., and Zhang K. A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles. IEEE Transactions on Intelligent Transportation Systems, 22(7):3975–3986, jul 2021. ISSN 1524-9050. DOI: 10.1109/TITS.2020.3002712.
31. Novikova E., Doynikova E., and Golubev S. Federated learning for intrusion detection in the critical infrastructures: Vertically partitioned data use case. Algorithms, 15(4), 2022. ISSN 1999-4893. DOI: 10.3390/a15040104.
32. Saputra F.A., Salman M., Hasim J.N., Nadhori I.U., Ramli K. The next-generation NIDS platform: Cloud-based snort NIDS using containers and big data. Big Data and Cognitive Computing, 6(1), 2022. ISSN 2504-2289. DOI: 10.3390/bdcc6010019.
33. Gong C., Zheng Z., Wu F., Shao Y., Li B., and Chen G. To store or not? online data selection for federated learning with limited storage. Proc. of the ACM Web Conference 2023, WWW '23, page 3044–3055, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394161. DOI: 10.1145/3543507.3583426.
34. Jiang C., Xia C., Liu Z., and Wang T. Feddroidmeter: A privacy risk evaluator for fl- based android malware classification systems. Entropy, 25(7), 2023. ISSN 1099-4300. DOI: 10.3390/e25071053..
35. Blanchard P., El Mhamdi E.M., Guerraoui R., and Stainer J. Machine learning with adversaries: Byzantine tolerant gradient descent. Proc. of the 31st International Conference on Neural Information Processing Systems, NIPS'17, pp.118–128, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.
36. Yin D., Chen Y., Kannan R., and Bartlett P. Byzantine-robust distributed learning: Towards optimal statistical rates. Proc. of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pp.5650–5659. PMLR, 10–15 Jul 2018.
37. Cao X., Fang M., Liu J., and Gong N.J. Fltrust: Byzantine-robust federated learning via trust bootstrapping. CoRR, abs/2012.13995, 2020. URL: https://arxiv.org/abs/2012.13995. (accessed on: 20.08.2023).
38. Flower — a friendly framework for federated learning. URL https://flower.dev/. (accessed on: 20.08.2023).
39. Li X., Jiang M., Zhang X., Kamp M., and Dou Q.. Fedbn: Federated learning on non-iid features via local batch normalization. CoRR, abs/2102.07623, 2021. (accessed on: 20.08.2023).
40. Reddi S.J., Charles Z., Zaheer M., Garrett Z., Rush K., Konecny J., Kumar S., and McMahan H.B. Adaptive federated optimization. CoRR, abs/2003.00295, 2020. (accessed on: 20.08.2023).
41. Yin D., Chen Y., Ramchandran K., Bartlett P.L. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. International Conference on Machine Learning. 2018. URL: https://api.semanticscholar.org/CorpusID:3708326 (accessed on: 20.08.2023).
42. Novikova E., Fedorchenko E., Kotenko I.., Kholod I. [Analytical Review of Intelligent Intrusion Detection Systems Based on Federated Learning: Advantages and Open Challenges] Аналитический обзор подходов к обнаружению вторжений, основанных на федеративном обучении: преимущества использования и открытые задачи. Informatics and Automation [Информатика и автоматизация]. 2023. No 22 (5). pp.1034–1082. DOI: DOI:10.15622/ia.22.5.4 (in Russian)