РАЗРАБОТКА ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА МОДЕЛИРОВАНИЯ МНОГОЗНАЧНЫХ КОМПЬЮТЕРНЫХ АТАК

Шелухин О. И.[1], Раковский Д. И.[2]

DOI: 10.21681/2311-3456-2024-4-116-130

Цель исследования: *разработка и программная реализация экспериментального программно-аппаратный комплекса (ПАК) для сбора телеметрии компьютерных сетей (КС) в условиях проведения многозначных контролируемых компьютерных атак (КА), а также анализ результатов имитационного моделирования многозначных атак, полученных с помощью реализованного комплекса.*

Методы исследования: *имитационное моделирование; машинное обучение; методы многозначного анализа; программная реализация программно-аппаратного комплекса для исследования свойства многозначности классовых меток.*

Объектами исследования *являются теоретические и практические вопросы многозначности классовых меток в сфере информационной безопасности.*

Результаты исследования. *Создан ПАК для сбора телеметрии в ходе имитационного моделирования компьютерных атак в компьютерных системах, обладающих свойством многозначности в табличном представлении. ПАК имитирует реальные данные, соответствующие задачам информационной безопасности. Новизна разработанного ПАК заключается автоматизированной параллельной маркировке всех КА, осуществляемых на КС, что позволяет учесть многозначность уже на этапе сбора данных. С использованием разработанного ПАК, сформирован многозначный набор данных, представляющий собой диагностическую информацию о сети, подвергаемой 3 типам КА, совершаемым параллельно – «Отказ в обслуживании»; «Сетевая разведка»; «Фаззинг». Обнаружено, что многозначная КА (совокупность однозначных КА) является отдельной сущностью с собственным распределением информативной значимости атрибутного пространства. Поскольку многозначная КА является отдельной сущностью, эта сущность может быть выявлена алгоритмами МО с высокой обобщающей способностью, способными к кластеризации. Если алгоритм МО не подразумевает многозначный выход, то даже при правильно выделенном кластере «внутри», отсутствие многозначного выхода приводит к ошибке классификации.*

Научная и практическая значимость. *Описан функционал предлагаемого ПАК для моделирования многозначных КА; формат данных в табличном представлении, собираемых при помощи разработанного ПАК. Данные, порождаемые ПАК, могут быть использованы при разработке средств обнаружения вторжений, учитывающих многозначность классовых меток. Предлагаемый ПАК позволяет исследовать свойство многозначности классовых меток посредством точной настройки соотношения однозначных и многозначных классовых меток за счет конфигуратора КА.*

Ключевые слова: *информационная безопасность, сетевые атаки, многозначная классификация, машинное обучение, имитационное моделирование, набор данных, экспериментальные данные.*

DEVELOPMENT OF A HARDWARE AND SOFTWARE SYSTEM FOR MODELLING MULTI-LABELED COMPUTER ATTACKS

Sheluhin O. I.[3], Rakovskiy D. I.[4]

The aim of the study: *development and software implementation of an experimental hardware-software complex for collecting telemetry of computer networks under conditions of multi-labeled controlled computer attacks, as well as analysis of the results of simulation modelling of multi-labeled attacks obtained with the help of the implemented complex.*

1 Шелухин Олег Иванович, доктор технических наук, профессор Московского технического университет связи и информатики, Москва, Россия. E-mail: sheluhin@mail.ru, ORCID: https://orcid.org/0000-0001-7564-6744

2 Раковский Дмитрий Игоревич, аспирант Московского технического университета связи и информатики, Москва, Россия. E-mail: Prophet_alpha@mail.ru, ORCID: https://orcid.org/0000-0001-7689-4678

3 Oleg I. Sheluhin., Dr.Sc., Full Professor, Moscow Technical University of Communications and Informatics, Moscow, Russia. E-mail: sheluhin@mail.ru, ORCID: https://orcid.org/0000-0001-7564-6744

4 Dmitry I. Rakovskiy, Postgraduate student, Moscow Technical University of Communication and Informatics, Moscow, Russia. E-mail: Prophet_alpha@mail.ru, ORCID: https://orcid.org/0000-0001-7689-4678

Research methods: *simulation modelling; machine learning; methods of multi-value analysis; software implementation of the hardware-software complex for the study of the property of multi-value class labels.*

Research results. *Objects of research are theoretical and practical questions of multi-labeled class labels in the sphere of information security.*

Scientific significance. *A hardware-software complex for telemetry collection in the course of simulation modeling of computer attacks in computer systems with multi-label property in tabular representation is created. hardware-software complex simulates real data corresponding to the tasks of information security. The novelty of the developed hardware-software complex is the automated parallel labeling of all computer attacks carried out on the computer network, which allows to take into account multi-label already at the stage of data collection. Using the developed hardware-software complex, multi-label data set is formed, which is diagnostic information about the network, subjected to 3 types of computer attacks made in parallel – «Denial of Service»; «Network Intelligence»; «Fuzzing». It is found that multi-label computer attack is a separate entity with its own distribution of informative significance of attribute space. Since multi-label computer attack is a separate entity, this entity can be detected by machine learning algorithms with high generalization ability capable of clustering. If the machine learning algorithm does not involve multi-label output, then even with a correctly identified cluster «inside», the lack of multi-label output leads to a classification error.*

Scientific and practical significance. *The functionality of the proposed hardware-software complex for modeling multi-label computer attacks is described; the format of data in tabular representation, collected with the help of the developed hardware-software complex. The data generated by the hardware-software complex can be used in the development of intrusion detection tools that take into account multi-label class labels. The proposed hardware-software complex allows to investigate the multi-label property of class labels by fine-tuning the ratio of single-valued and multi-label class labels through the computer attack configurator.*

Keywords: *Information security, network attacks, multi-label classification, machine learning, simulation modeling, dataset, experimental data.*

Введение

Системы обнаружения вторжений (СОВ), в чьей основе находятся алгоритмы машинного обучения (МО), как правило, требуют объемной выборки «исторических данных», соответствующих защищаемой компьютерной сети (КС) [1–3]. Для корректной работы СОВ, «исторические данные» должны содержать актуальные типы компьютерных атак (КА); реализации каждой КА должны быть разнообразными по своим параметрам[4].

Одной из актуальных особенностей данных, влияющих на качество решения задач классификации и прогнозирования, является многозначность классовых меток[5] [5]. Исследованию свойства многозначности посвящен ряд работ, связанных с медициной, компьютерным зрением, работой с текстом [6,7]. Учет многозначности классовых меток позволяет снизить количество ложноотрицательных и ложноположительных ошибок классификации [8].

Как правило, существующие наборы данных, описывающие поведение КС в момент совершения КА, игнорируют многозначность данных, как, например, UNSW-NB15 [9]. Анализ существующих наборов данных, находящихся в открытом доступе, показал, что многозначные наборы данных, пригодные для решения задач многозначной классификации по ряду вопросов информационной безопасности, либо отсутствуют, либо доля многозначных записей ничтожна, что требует создания специализированного стенда для целенаправленного формирования многозначных данных в контролируемых условиях. Редким исключением является многозначная база данных SR-BH 2020 [10].

Редкость баз данных, содержащих многозначные КА, находящихся в открытом доступе, актуальной является разработка и реализация программно-аппаратного комплекса (ПАК) для сбора телеметрии и имитационного моделирования многозначных КА. Важным условием функционирования ПАК является проведение каждой КА в контролируемых условиях.

Анализ существующих решений в сфере ИБ [11] (а также см.[6]) выявил уникальность предлагаемого решения: не существует программных или программно-аппаратных решений, находящихся в открытом доступе, направленных на исследование свойства многозначности в данных.

Целью работы является разработка и программная реализация экспериментального ПАК для сбора телеметрии КС в условиях проведения многозначных контролируемых КА, а также анализ результатов имитационного моделирования многозначных атак, полученных с помощью реализованного комплекса.

Структурная схема функционирования ПАК

Формализуем механизм работы ПАК для сбора телеметрии и имитационного моделирования

5 Gibaja E., Ventura S. A Tutorial on Multilabel Learning // ACM Comput. Surv. 2015. Т. 47, № 3. С. 1–38с. DOI: 10.1145/2716262

6 Д. И. Котенко, И. В. Котенко, И. Б. Саенко, Методы и средства моделирования атак в больших компьютерных сетях: состояние проблемы, Тр. СПИИРАН, 2012, выпуск 22, 5–30

многозначных КА. Зададим топологию T исследуемой КС в виде двух множеств хостов:

$$T = \{VH_i;\ i = \overline{1,I}\} \cup \{AH_j;\ j = \overline{1,J}\} \cup DAS \cup Router, \quad (1)$$

где VH_i – i-й хост, имитирующий жертву (далее – атакуемый хост, от англ. *Victim host*); AH_j – j-й хост, имитирующий машину злоумышленника (далее – атакующий хост, от англ. *Attack host*), проводящую контролируемую КА на VH_i; DAS – сервер агрегации данных (англ. *data aggregation server*), аккумулирующий телеметрию с VH_i и AH_j, а также содержащий конфигуратор КА; $Router$ – маршрутизатор (группа маршрутизаторов, или фрагмент сети Интернет), соединяющий множество атакуемых и атакующих хостов.

В контексте разработки ПАК уместно говорить о проведении контролируемых компьютерных атак (ККА), чье проведение полностью прогнозируемо на этапе планирования и контролируется в течение хода эксперимента. Введем в рассмотрение перечень ККА AL, которые атакующие хосты AH_j способны реализовать на атакуемые хосты VH_i:

$$AL = \{attackk;\ k = \overline{1,K}\}. \quad (2)$$

Каждая КА описывается рядом статичных $attack_k$ и варьируемых $vattack_k$ параметров – AoI_k:$attack_k \cup vattack_k$. Статичные параметры $attack\_k$: $\langle params_{l_k}\rangle$; $pl_k = \overline{1,PL_k}$ являются общими для каждой реализации КА. Общее число параметров атаки PL_k и их содержательное наполнение варьируется в зависимости от специфики КА[7].

Введем варьируемые параметры АК, которые могут изменяться в рамках конкретной реализации – AoI_k (англ. *Attack on Interval*). Такие параметры задаются либо фиксированными числами, либо законами распределения, выбираемыми из библиотеки распределений $FL = \{F_{lf}(\alpha_p);\ p = \overline{1,P_{lf}},\ lf = \overline{1,LenF}\}$, где α_p – p-й параметр lf-го закона распределения:

$$AoI_k\langle IS,IE,attack_k,ah,tar,dur,int,etcp\rangle, \quad (3)$$

где IS – параметр, точное время начала интервала атаки; IE – параметр, точное время окончания интервала атаки; ah – атакующий хост, реализующий экземпляр атаки в пределах указанных интервалов; tar – множество целей атаки для dur – длительность атаки в пределах указанных интервалов; int – интенсивность атаки в пределах указанных интервалов; $etcp$ – множество иных варьирующихся параметров.

Конкретное множество параметров $etcp$ детализируется в зависимости от механизма реализации КА. Общее число параметров атаки – Lk – варьируется в зависимости от требуемой детализации. Значения параметров обусловлено механизмом реализации атаки, сложностью ее исполнения, используемыми протоколами, приложениями, устройствами.

При практической реализации сборщика, необходимо учесть условия функционирования анализируемой КС. Поскольку анализируемая КС является целью проведения КА, в ней допускается нарушение целостности, доступности и конфиденциальности информации, а также деструктивное воздействие на поддерживающую инфраструктуру T.

При превышении некоторого критического порога деструктивного воздействия на систему КС выходит из строя и сбор телеметрии с нее становится невозможным. Для предотвращения уничтожения КС в следствие фатального воздействия КА, предусмотрен механизм оценки максимально допустимого негативного воздействия на КС i-й атакуемый хост VH_i – $MaxDamage_{VH_i}$ – со стороны атакующих хостов. Под $MaxDamage_{VH_i}$ будем понимать максимально допустимое время ответа i-го атакуемого хоста VH_i на синхронизирующий сигнал, поступающий с сервера агрегации данных DAS.

Взаимодействие между элементами КС DAS, VH_i и AH_j (4) осуществляется через программные агенты 1-го и 2-го типов, распространяемые на соответствующие хосты: $PA = \{prograg_{1,i};\ i = \overline{1,I}\} \cup \{prograg_{2,j};\ j = \overline{1,J}\}$.

Программные агенты обоих типов связаны с сервером агрегации данных DAS. Программные агенты 1-го типа $prograg_{1,i}$ осуществляют сбор телеметрической информации с атакуемых хостов VH_i и их передачу на DAS. Программные агенты 2-го типа – $prograg_{2,j}$ – осуществляют сбор телеметрической информации с атакующих хостов AH_j и реализуют КА, связанные с AH_j, согласно управляющим командам, поступающим с DAS.

Так как технические показатели хостов топологии T (2) могут отличаться, то выбрать одинаковую частоту сбора телеметрической информации для все хостов не представляется возможным. Неверный выбор частоты сбора телеметрической информации может повлечь за собой излишнюю нагрузку на вычислительные мощности атакуемых хостов, что критично при проведении ККА из-за угрозы превышения максимально допустимого времени ответа i-того атакуемого хоста VH_i – $MaxDamage_{VH_i}$.

Одним из способов вычисления допустимой частоты сбора телеметрической информации является запуск нагрузочного тестирования на каждом VH_i и вычисление среднего времени, необходимого для обработки одной итерации сбора телеметрии.

На этапе проведения ККА, в момент реализации атаки, с помощью датчика псевдослучайных чисел

7 CAPEC – Common Attack Pattern Enumeration and Classification (CAPECTM) [Электронный ресурс]. URL: https://capec.mitre.org/index.html (дата обращения: 12.09.2023).

RND формируется закон распределения $F_{params_{lk}}$ с выбранными параметрами реализации атаки внутри каждого интервала $IS - IE$. В случае необходимости полного контроля за осуществлением ККА параметры распределения заменяются фиксированными числами.

Взаимодействие между атакующими хостами AH_j и хостами-жертвами VH_i, описывается вектором:

$$AoI_k: \vec{V}_k = (AoI_{kw}; w = \overline{1,W_k}), \quad (4)$$

где W_k – количество случаев, когда k-я КА $attack_k$ реализуется в течение эксперимента.

В рамках топологии (1) результате воздействия атакующими хостами AH_j на хосты-жертвы VH_i компьютерными атаками (2) с параметрами (3), объединенными в вектора (4), формируется нагрузка на атакуемые хосты, считываемая программными агентами 1-го типа и отравляемая на сервер агрегации данных.

Конфигурация воздействия по каждой КА (расписание КА) может быть представлена в виде итогового множества CoA (англ. *Chronology of Attacks*):

$$CoA = (\vec{V}_k; k = \overline{1,K}). \quad (5)$$

Визуализация формализованного выше приведенными соотношениями механизма работы стенда, представлена на рис. 1.

Топология T для исследуемой КС отражена путем визуализации в виде двух контролируемых зон – зоны атакуемых VH_i и атакующих AH_j хостов. Зоны VH_i и AH_j, в свою очередь, разделены неконтролируемой зоной, имитирующей сеть Интернет и содержащей маршрутизатор *Router*.

На каждом из VH_i установлен программный агент первого типа $prograg_{1,i}$. На каждом из хостов AH_j установлен программный агент второго типа $prograg_{2,j}$. На сервере агрегации данных DAS расположена база данных для агрегируемый телеметрических данных с AH_j и VH_i. DAS предназначен для контроля взаимодействия между программными агентами. Маршрутизатор сети является связующим звеном между всеми хостами VH_i и AH_j.

Программные агенты второго типа – $prograg_{2,j}$ – распространяются на атакующие хосты AH_j; их задачами является:

- Взаимодействие с DAS с целью получения расписания ККА;
- Проведение ККА согласно полученному расписанию;
- Отправка на DAS информации об успешном старте и остановке проведения ККА согласно полученному расписанию;
- Обмен диагностической информацией с DAS.

Для реализации КА сервер агрегации данных DAS посылает управляющие сигналы на атакующие хосты. Реализация каждой КА на атакующем хосте выполняется в виде вызываемого docker-контейнера, содержащего предустановленное программное

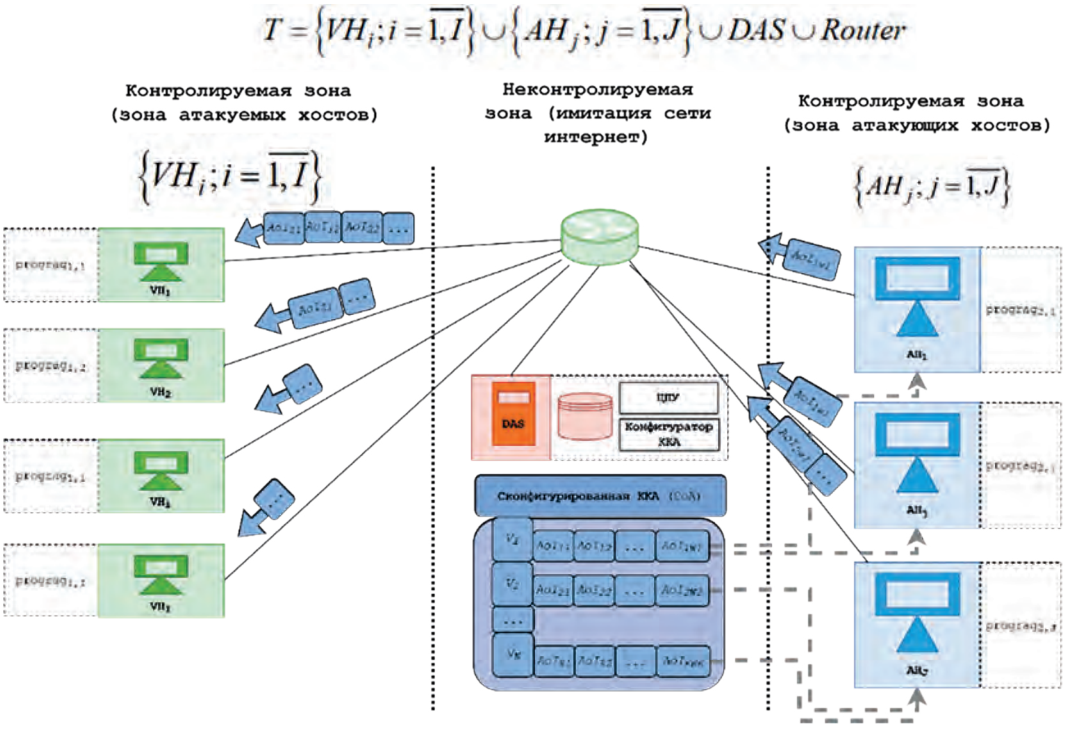

Рис. 1. Структурная и схема функционирования ПАК

обеспечение и скрипты для выполнения КА. Управление ходом КА (ее прекращение или повторение заданное количество раз) осуществляется программным агентом $prograg_{2,j}$.

В качестве иллюстрации на рис. 1, приведены несколько элементов CoA, направленных на различные атакуемые хосты (детализация дана для хостов VH_1, VH_2). На хост VH_1 направлены атаки двух типов: две реализации атаки AoI_{11}, AoI_{12}, и реализация атаки AoI_{22}. На хост VH_2 направлена реализация атаки AoI_{22}. Полная информация о сконфигурированных компьютерных атаках – CoA – доступна на DAS, в конфигураторе ККА. Детализация данного узла раскрывается в соответствующем разделе.

Данные телеметрии собираются программными агентами на DAS. Здесь же производится маркировка и последующая аккумуляция данных, поступающих с программных агентов. При необходимости из сформированной базы данных осуществляется выгрузка дампов в пригодном для последующего анализа выбранными алгоритмами МО.

Сценарий использования разработанного ПАК, при известном перечне КА $AL = \{attack_k;\ k = \overline{1,K}\}$ и их статичных параметрах $\langle params_{l_k}\rangle$, включает:

1) настройку взаимодействия атакуемых хостов VH_i между собой в рамках эксперимента (определение роли каждого хоста в рамках моделируемого бизнес-процесса; актуализация программного обеспечения; сетевой топологии на хостах и на $Router$);
2) развертывание подсети атакующих хостов AH_j;
3) создание «расписания КА»: задание векторов AoI_k по каждой из K КА при помощи разработанного конфигуратора КА;
4) инициализацию эксперимента: запуск программных агентов 1-го и 2-го типов для сбора данных; проверка корректности их взаимодействия с DAS; запуск стороннего программного обеспечения для сбора дополнительной телеметрии (при необходимости) на VH_i;
5) проведение эксперимента: КА реализуются атакующими хостами AH_j согласно управляющим командам, посылаемым с DAS на программные агенты 2-го типа;
6) завершение эксперимента и формирование выходных данных.

После завершения эксперимента, ПАК формирует многозначный набор данных, содержащий диагностическую информацию о сети, подвергаемой КА из перечня $AL = \{attack_k;\ k = \overline{1,K}\}$ согласно п. 3 сценария. ПАК включает в себя ряд скриптов, написанных на языке python, позволяющих объединить диагностическую информацию, собранную с программных агентов и сторонних сборщиков телеметрии: *Wireshark, MSI Afterburner, Windows Perfmon*.

Особенности имитационного моделирования многозначных КА в ПАК

Для тонкой настройки проведения серии ККА с различными параметрами на атакуемые хосты VH_i, в ПАК реализован конфигуратор ККА, содержащий библиотеку статичных параметров $params_{l_k}$; библиотеку распределений случайных величин $F_l(\alpha_p)$, используемых при формировании варьируемых параметров AoI_k во время проведения эксперимента; функционал планировщика ККА и связанного с ним расписания проведения ККА.

Конфигуратор ККА необходим для планирования и автоматизации объемных во времени экспериментов. Он позволяет задать точное время начала и конца каждой ККА из перечня доступных для реализации. Визуализация конфигуратора ККА приведена на рис. 2.

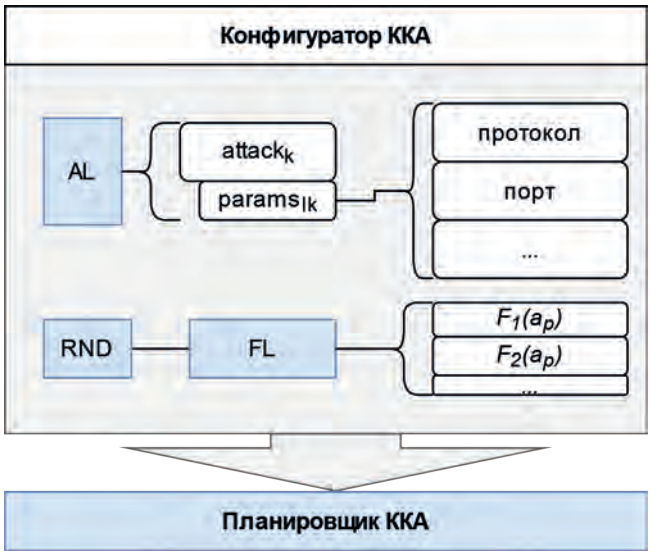

Рис. 2. Визуализация конфигуратора ККА

Согласно (2) .... (4), КА каждого типа характеризуется ее параметрами $params_{l_k}$, общими для каждой отдельной реализации такой атаки, и рядом варьируемых параметров AoI_k:$\langle IS,IE,attack_k,ah,tar,dur,int,etcp\rangle$, уникальных для каждой реализации КА.

Набор параметров $params_{l_k}$ задается перед началом эксперимента и в дальнейшем не меняется. Содержимое $params_{l_k}$, как правило, уникален для каждого типа КА и прописывается в bash-скриптах, находящихся в вызываемых контейнерах *Docker*, реализующих механизм указанной атаки.

Содержимое AoI_k задается либо константами, либо законами распределения – $F_l(\alpha_p)$. Формирование случайных величин в соответствии с $F_l(\alpha_p)$ происходит с помощью генератора псевдослучайных чисел по законам распределения из библиотеки распределений FL.

Планировщик ККА является важным элементом конфигуратора ККА. Временная диаграмма, иллюстрирующая работу планировщика ККА, приведенная на рис. 3, позволяет наглядно визуализировать многозначные классовые метки [12]. Каждая реализация атаки задается параметрами – AoI_k (5). Для задания итогового множества CoA (5), необходимо воспользоваться планировщиком ККА. На рисунке 3, по оси абсцисс отложено время эксперимента, в рамках которого осуществляются ККА. В качестве иллюстрации перечня возможных КА, отложенных на оси ординат, выбраны атаки из базы данных *CAPEC* [13]:

- две атаки типа «отказ в обслуживании», направленные на один хост исследуемой сетевой топологии (*CAPEC-125: Flooding*) – DoS_{p1} и DoS_{p2};
- атака типа «сканирование портов» (*CAPEC-300: Port Scanning*);
- атака типа Scanning for Vulnerable Software (*CAPEC-310: Scanning for Vulnerable Software*).

Поскольку реализуется несколько КА типа «отказ в обслуживании», направленных на хост исследуемой системы, в моменты одновременной реализации данных атак справедливо говорить об атаке типа «массовый отказ в обслуживании».

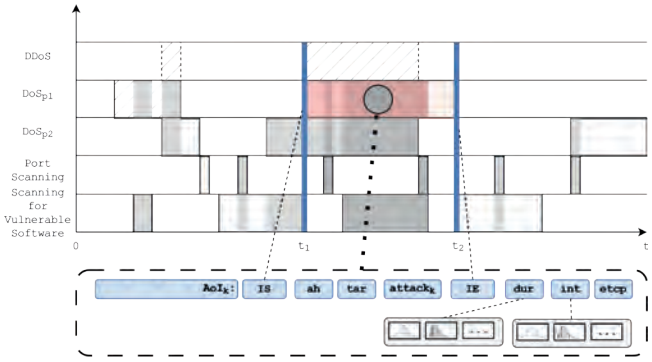

Рис.3. Временная диаграмма визуализации работы механизма планирования ККА

Диаграмма содержит прямоугольные области, отмечающие точное время начала и окончания интервала каждой компьютерной атаки. В рамках каждого интервала задаются параметры КА. В качестве иллюстрации, приведена детализация параметров КА типа «отказ в обслуживании». В пределах каждого интервала определены варьируемые параметры AoI_k:⟨*IS*,*IE*,$attack_k$,*ah*,*tar*,*dur*,*int*,*etcp*⟩. Ряд метрических параметров (таких, как *dur* и *int*) может быть задан как константой, так и законом распределения из библиотеки *FL*.

Многозначность собираемых данных проиллюстрируем на конкретном примере. Для этого рассмотрим визуализацию расписания ККА. *Визуализация расписания ККА* необходима для облегчения планирования ККА человеком и контроля за долей многозначных КА в формируемом наборе данных. Пример визуализации расписания ККА для одного атакуемого хоста приведен в табл. 1. Каждые сутки в таблице представлены 24 ячейками, маркированными от «00:00» – полуночи до «23:00» – одиннадцати часов вечера. Интервал времени – час. В каждый из интервалов времени может быть реализована одна или несколько КА (в визуализации – 6 типов). Параметры каждой КА (в том числе и атакующий хост, IP-адрес атакуемого хоста, интенсивность КА и т.д.) задаются в конфигураторе КА; на представленной визуализации данная информация опускается для большей наглядности.

В таблице присутствует цветовое разделение, выполненное в виде градаций серого цвета:

- Белым цветом отмечены интервалы, когда КА определенного типа не реализуется.
- Светло-серым цветом отмечены интервалы, когда КА реализуется и при этом кроме данной КА

Таблица 1.

Пример расписания КА для одного атакуемого хоста

Дата	Время, ч	Атака №1	Атака №2	Атака №3	Атака №4	Атака №5	Атака №6
22.04.2024	0:00						
	1:00						
	2:00						
	3:00						
	4:00						
	5:00						
	6:00						
	7:00						
	8:00						
	9:00						
	10:00						
	11:00						

в данном интервале ни одна другая КА не реализуется (однозначные КА);

- Темно-серым цветом отмечены интервалы, когда КА реализуется и при этом реализуется иная КА – наблюдается многозначная КА.

Примерами интервалов времени, когда наблюдаются многозначные КА, являются:

- 22.04.2024; 0:00 – 2:00 (КА №1 и №4);
- 22.04.2024; 2:00 – 5:00 (КА №1, №2 и №4);
- 22.04.2024; 7:00 – 8:00 (КА №1, №2, №3);
- и иные.

Поскольку в моменты производится воздействие сразу нескольких КА с разными параметрами на один атакуемый хост VH_i, их совокупное *синергетическое* воздействие может приводить к фатальным последствиям для последнего [14]. В результате такого воздействия, происходит одновременное «наложение» реализаций атак (как это также отмечено на рис. 3; интенсивность цвета отражает количество одновременно воздействующих КА). Представленная многозначность данных наблюдается в реальных компьютерных системах [8,15].

Результатом работы ПАК является сформированная многозначная база данных КА, предназначенная для исследования специфического явления – многозначности классовых меток компьютерных атак.

Конфигуратор ККА допускает настройку стенда на однозначный–бинарный или многоклассовый режимы работы. Для реализации бинарного режима работы стенда вида «нормальное состояние КС – реализация конкретной атаки» в конфигураторе необходимо выбрать один тип атаки, после чего настроить интервалы ее реализации.

Для реализации многоклассового режима работы ПАК в конфигураторе предусмотрено разграничение интервалов начала и окончания атак каждого типа строго без показанных выше пересечений по времени.

Отметим, что при планировании эксперимента с помощью ПАК, рекомендуется ориентироваться на эксплуатационные характеристики телекоммуникационного оборудования, используемого для имитации сети Интернет (узел *Router* в топологии T). При выходе из строя узла, связывающего подсеть атакующих хостов и атакуемых хостов, ряд запланированных в расписании ККА производится не будет до устранения неисправности на маршрутизаторе. Во избежание потери данных, в ПАК предусмотрено резервирование данных на каждом хосте, на котором располагаются программные агенты. В случае выхода из строя узла (ряда узлов) сети ПАК, возможно восстановление данных из резервных копий.

Анализ многозначных данных, формируемых ПАК

Для решения задач классификации, прогнозирования и исследования многозначных данных в ПАК программно реализован новый исследовательский фреймворк (ИФ). Фреймворк представляет собой шаблон, облегчающий сравнение алгоритмов МО между собой в задачах прогнозирования и классификации. Архитектура разработанного исследовательского фреймворка (ИФ), реализованна на Python версии 3.10, с применением следующих открытых библиотек: *pandas, seaborn, matplotlib, time, numpy, sklearn, keras, tensorflow*. Предусмотрено сравнение алгоритмов МО в задачах бинарной, однозначной и многозначной классификаций.

Процесс проведения исследования можно разделить на **два этапа**:

Этап 1 *– исследование и предобработка исходных экспериментальных данных;*

Этап 2 *– проведение эксперимента классификации.*

Этап 1. Процесс исследования свойств экспериментальных данных начинается с задания исходных параметров предобработки ЭД табличного типа. В качестве входных данных на ИФ подается:

- Переменная, отвечающая за тип классификации: бинарная, многоклассовая, многозначная.
- Логическая переменная, отвечающая за необходимость предварительного перемешивания данных.
- Логическая переменная, отвечающая за необходимость трансформации атрибутов ЭД.
- Логическая переменная, отвечающая за необходимость формирования ROC-кривых.
- Логическая переменная, отвечающая за метод построения ROC-кривой: «Один против одного» (One-vs-one, OVO) или «один против всех» (One-vs-everyone, OVE или One-vs-rest – OVR).
- Наименование эксперимента. В наименование эксперимента обязательно включается информация обо всех логических переменных, содержащихся в исходных данных.
- Количество блоков разделения ЭД в режиме перекрестной проверки (кросс-валидации) по нотации *K-Fold*. По умолчанию используется «классическая» кросс-валидация с разделением исходных ЭД на два блока: блок обучающих данных и блок тестовой выборки.
- Массив, содержащий в себе наименование всех вторичных атрибутов, исследуемых ЭД.
- Переменная, отвечающая за тип эксперимента, проводимого на этапе 2.
- Набор переменных для оптимизации вычислений: логическая переменная, отвечающая за необходимость пропуска этапа 1 в случае наличия

заранее обработанной информации; переменная, ограничивающая количество циклов, выполняемых на этапе 2 и так далее.

Визуализация этого этапа приведена на рис. 4.

Этап 1 состоит из пяти *шагов*, на каждом из которых происходит обработка данных и формирование сопутствующих выкладок: таблиц, графиков, диаграмм, текстовой информации и прочего. Рассмотрим данные шаги более подробно.

Шаг 1. Получение первичных данных и разведочный анализ. Первичные данные выгружаются в ИФ из таблицы формата .csv (*пункт (1) на схеме рис. 4*), после чего выполняется их разведочный анализ.

Разведочный анализ данных представляет собой анализ ЭД по каждому атрибуту $A_m = \{a_{mn};\ m = \overline{1,M},\ n = \overline{1,N}\}$ по следующим показателям: количество записей (count): $count_{A_m} = |A_m| = N$; Среднее (mean): $mean_{A_m} = \bar{A}_m$; Среднее квадратическое отклонение (*corrected sample standard deviation* (в нотации библиотеки pandas – *sample standard deviation*, STD) – $STD_{A_m} = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(a_{mn} - \bar{A}_m)^2}$; минимальное значение атрибута (min): $min_{A_m} = minA_m$; максимальное значение атрибута (max) $max_{A_m} = maxA_m$; нижний, 50%-й и верхний процентили (percentile, P). Результаты разведочного анализа маркированы *пунктом (2) на схеме рис. 4*.

Дополнительно оценивается количество уникальных значений у атрибута: $\mu_m{:}A_m \longrightarrow N \cup \{0\}$, где – множество натуральных чисел с включением нуля. Также оценивается количество некорректных (NaN, пропусков) значений атрибута. Результат разведочного анализа сводится в таблицу и сохраняется. Этому пункту соответствует таблица с обозначенными результатами анализа (*пункт (3) на схеме рис. 4*). Дополнительно выполняется частотный анализ целевого столбца, данные выводятся в отдельную таблицу и сохраняются для последующей визуализации.

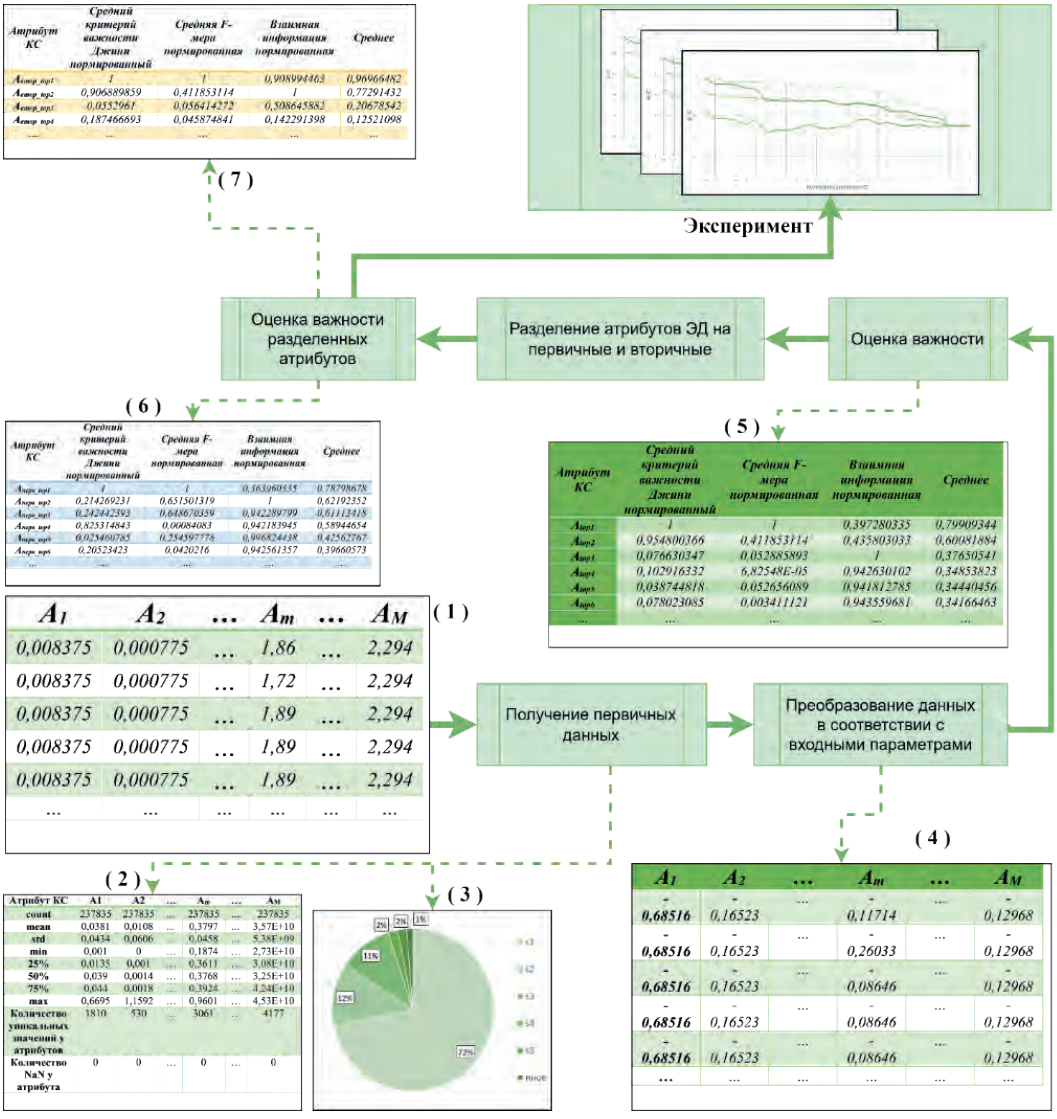

Рис.4. Архитектура ИФ. Этап предобработки данных

Шаг 2. Преобразование данных в соответствии с входными параметрами. На данном шаге ЭД анализируются по критерию «количество уникальных значений атрибута». Если «количество уникальных значений атрибута» равно единице – атрибут исключается из дальнейшего анализа.

Если логическая переменная, отвечающая за необходимость предварительного перемешивания данных равна истине – то данные перемешиваются в режиме «группировка по одной записи». Для сохранения возможности повторного проведения эксперимента начальное значение генератора случайных чисел всегда ставится равной определенной константе.

Если логическая переменная, отвечающая за необходимость трансформации атрибутов ЭД, равна «истине» – то проводится нормализация и стандартизация атрибутов посредством удаления среднего значения и масштабирования всех атрибутов до единичной дисперсии [16]. Кроме изложенного на данном шаге, выполняется кодирование категориальных меток классов под стандарты классификаторов scikit-learn. Категориальные атрибуты кодируются своими порядковыми номерами, например, «а» соответствует «1»; «б» соответствует «2» и так далее. Порядковый номер присваивается в порядке первого вхождения метки класса в ЭД. Отметим, что перекодирование никак не влияет на итоговые результаты классификации. Результатом выполнения данного шага является таблица трансформированных данных (*пункт (4) на схеме рис. 4*).

Шаг 3. Оценка важности атрибутов ЭД. На данном шаге выполняется оценка важности атрибутов ЭД по разным критериям. В реализованном ИФ реализованы три упомянутых метрики на базе библиотеки scikit-learn. Статистический критерий важности атрибутов вычисляется на основании p-value, взаимной информации. В критерии важности, вычисленные на основании *p-value*, включена *F-мера*, вычисленная между метками класса и значениями атрибутов посредством дисперсионного анализа (*ANalysis Of VAriance, ANOVA*). Реализован также способ вычисления F-меры, вычисленной между метками класса и значениями атрибутов методами регрессионных тестов.

Каждая из вышеприведенных оценок может быть вычислена различными мета-методами. Отметим, что для задач вывода важности всех атрибутов и их последующего отбора, не обязательно использовать все перечисленные мета-методы, поскольку сами по себе они не влияют на результаты оценки, а влияют лишь на способ отбора по существующим оценкам. Выбор мета-методов должен быть определен исследователем в контексте решаемой им задачи.

SelectKBest – метод формирования оценки и отбора k лучших атрибутов по определенной метрике: $Kfold_{sort(A,PARAM)} = \{A_{top1}, At_{op2}, ..., A_{topK} \mid K \le M\}$, где $sort(\boldsymbol{A}, PARAM)$ – функция сортировки по убыванию набора атрибутов $\boldsymbol{A}$ по некоторому критерию (оценке, параметру) – $PARAM$ – возвращающая набор атрибутов, K – параметр метода; A_{top1} – атрибут, имеющий наивысшую оценку среди всех атрибутов; At_{op2} – атрибут, ранжированный на второе место – и так далее. Метод SelectKBest, как следует из его названия, отбирает K атрибутов КС, имеющих наибольшую важность.

SelectPercentile – метод формирования оценки и отбора атрибутов по наивысшему перцентилю по определенной метрике

$$Kpercentile_{sort(A,PARAM)} = \\ = \{A\_topk \mid k = \overline{1,\lceil M \times 0.01 \times K \rceil}; K \le 100\}.$$

SelectFpr – метод формирования оценки и отбора атрибутов по наименьшему количеству ошибок (выражается через p-значение, α) первого рода по определенной метрике:

$$SelectFPR_{sort(A,PARAM),\alpha} = \{\forall A_k \mid pvalue\ A_k \le \alpha\}.$$

SelectFdr – метод формирования оценки и отбора атрибутов по наименьшему количеству ошибок второго рода, вычисляемая из p-value по определенной метрике:

$$SelectFDR_{sort(A,PARAM)} = \{\forall A_k \mid pvalue\ A_k \in BHP\},$$

где BHP – процедура Бенджамина-Хохберг (*Benjamini-Hochberg procedure*)[8].

SelectFwe – метод формирования оценки и отбора атрибутов по частоте ошибок по семействам (*Family-wise error rate*, FWE) по определенной метрике:

$$SelectFWER_{sort(A,PARAM),\alpha} = \{\forall A_k \mid 1 - FP_{TP=0} \le \alpha\}.$$

В случае необходимости выбора нескольких мета-алгоритмов оценки атрибутов по важности, в разработанном ИФ предусмотрена функция усреднения по группе. В каждой группе (Джини, p-value, взаимная информация) метрики, полученные при помощи перечисленных выше мета-методов, усреднялись и нормировались по трем группам. Результаты оценки значимости атрибутов визуализированы в виде таблицы, *пункт (5) на схеме рис. 4*.

Шаг 5. Разделение атрибутов ЭД на первичные и вторичные. На данном этапе в соответствии с входными параметрами происходит разделение атрибутов ЭД на первичные и вторичные. Первичные

8 Benjamini Y., Hochberg Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing // Journal of the Royal Statistical Society Series B: Statistical Methodology. 1995. Т. 57, № 1. С. 289–300. DOI: 10.1111/j.2517-6161.1995.tb02031.x

и вторичные атрибуты КС сохраняются в отдельные таблицы.

Шаг 6. Оценка важности разделенных атрибутов. Полученные наборы данных еще раз оцениваются по важности по указанным на шаге 3 критериям, внутри своих групп (*пункты (6) и (7) на схеме рис. 4*).

После предобработки ЭД, оценки важности их атрибутов, первичные атрибуты ЭД, повторно оцененные по важности внутри своей группы и подаются на вход Этапа 2.

Этап 2. На этапе 2 оцениваются результаты решения задачи классификации или прогнозирования. Предусмотрена «классическая» перекрестная проверка с разделением сходных ЭД на два блока: блок обучающих данных и блок тестовой выборки. Каждый шаг перекрёстной проверки разбивает исходные ЭД на несколько блоков равного объема; при этом один блок является тестовой выборкой ЭД, а остальные – обучающей.

Визуализация этого этапа приведена на рис. 5. Для корректной работы алгоритма атрибуты исходных экспериментальных данных оцениваются по информативности (*пункт (1) на схеме рис. 5*).

В ИФ реализована перекрестная проверка только по нотации *K-Fold* с разделением только на обучающую и тестовую выборки (без валидационной выборки). Метрики включают в себя как метрики оценки качества среднего по множеству классов (так называемые макро-метрики), так и метрики оценки качества на основе множества записей (микро-метрики). В дополнение к вычислению известных бинарных оценочных метрик на основе количества истинно положительных результатов (*TP*) истинно отрицательных результатов (*TN*), ложноположительных результатов (*FP*) и ложноотрицательных результатов (*FN*) вычисляется *accuracy*, *precision*, *recall f-мера*, *ROC* и связанная с ней *AUC*.

По окончанию цикла перекрестной проверки, из ЭД удаляется первичный атрибут в соответствии с некоторым условием. Условие может быть «максимальная важность» и «минимальная важность». После удаления одного из атрибутов циклы повторяются заново. Как только все атрибуты окажутся удаленными – ИФ завершит свою работу, породив таблицы итоговых результатов: детализированная таблица экспериментов по каждому блоку перекрестной проверки (*пункт (2) на схеме рис. 5*) и обобщенная (усредненная) таблица итогов проведенного эксперимента (*пункты (3) и (4) на схеме рис. 5*).

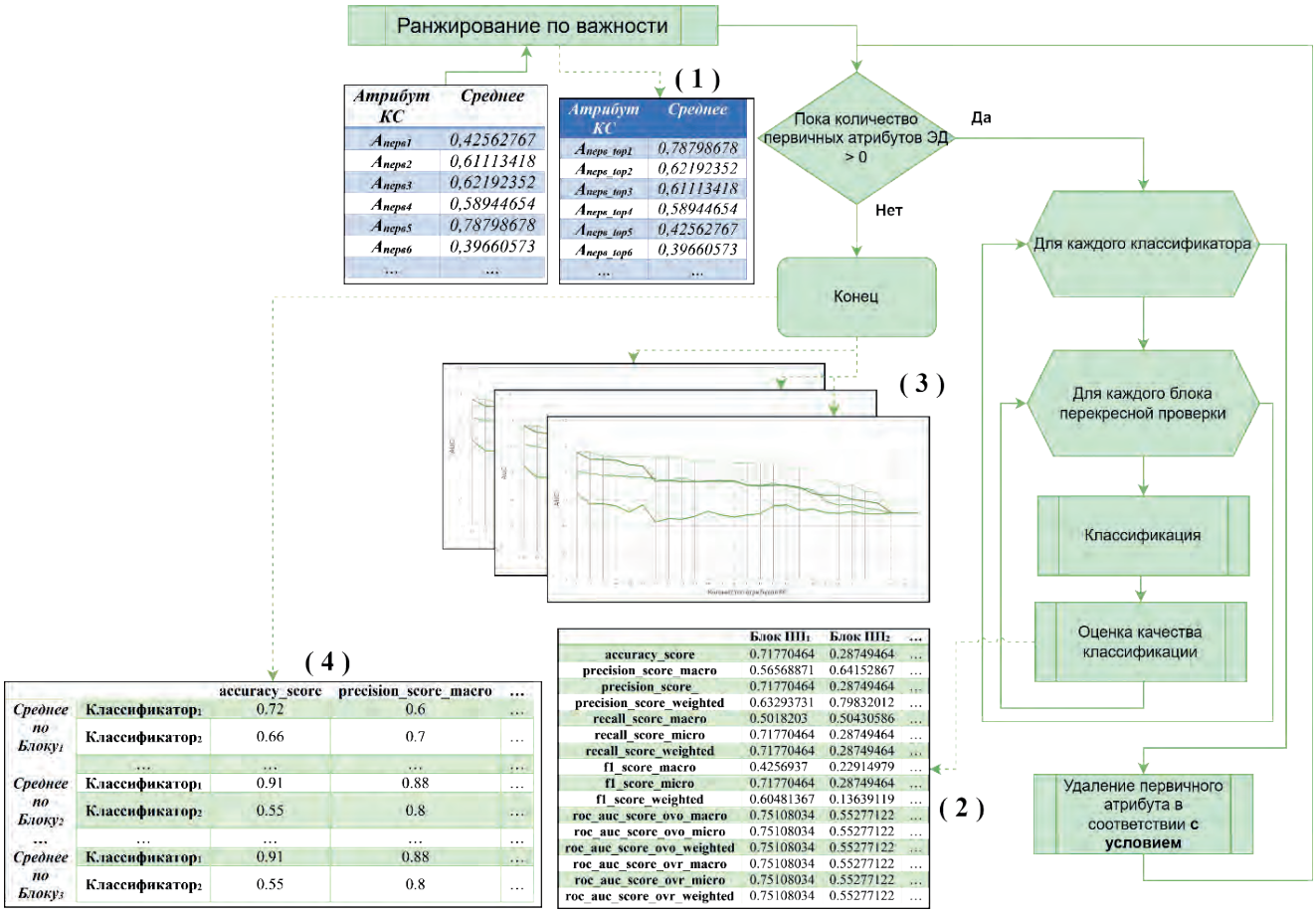

Рис. 5. Архитектура фреймворка. Этап проведения эксперимента

Исследование влияния сокращения атрибутной размерности на эффективность классификации по нескольким метрикам способствует решению следующих задач, сопряженных с анализируемым набором ЭД:

1. Сравнение разных методов классификации ЭД: бинарный, многоклассовый, многозначный.
2. Оценка степени влияния проблемы классового дисбаланса в частотной и временной областях на результаты классификации при различных подходах к предобработке исходных ЭД.
3. Оценка степени влияния проблемы атрибутной размерности («проклятие размерности»
4. Экспериментальная проверка влияния важных (незначительных) атрибутов ЭД на результаты классификации.
5. Сравнить результаты классификации на различных наборах ЭД.
6. Сравнение классификаторов, основанных на различных подходах, принципах и математических аппаратах, при различной атрибутной размерности.
7. Проверка эмпирической гипотезы о монотонности функции зависимости убывания эффективности классификации целевого столбца по какой-либо метрике.

Основными недостатками разработанного ИФ являются:

1) Зависимость времени исследования ЭД от реализаций алгоритмов в открытых библиотеках *pandas*, *seaborn*, *matplotlib*, *time*, *numpy*, *sklearn*.
2) Отсутствие графического интерфейса ИФ (работа с ИФ предполагает работу с программным кодом).

Результаты проведенного эксперимента с использованием предложенного ПАК

Количество записей экспериментальных данных в итоговом наборе составлял 263.388 шт. Набор данных содержал 125 атрибутов метрического типа и был разделен на 3 категории: аппаратные атрибуты атакуемого хоста; атрибуты, связанные с сетевым взаимодействием; атрибуты, извлеченные из системных журналов операционной системы Windows. Информация, собранная с сетевой карты и системных журналов, преобразовывалась в метрические атрибуты посредством вычисления количественных характеристик (средняя длина пакета; количество уникальных событий; количество уникальных сессий и т.д.) в рамках окна размером 1 секунда.

Проведённый анализ атрибутного пространства выявил, что атрибутами с наибольшим количеством уникальных значений являются атрибуты, извлеченные из журналов ОС *Windows*, а также атрибуты, связанные мониторингом функционирования видеокарты. Расписание проведения ККА сконфигурировано таким образом, чтобы часть атак различных категорий происходила одновременною. Реализация нескольких ККА одновременно в соответствии с расписанием позволил собрать многозначные данные.

Распределение атак с каждого атакующего хоста приведено на рис. 6.а. Распределение атак каждого типа приведено на рис 6.б. Подсчет классовых меток для построения частотной статистики в рамках указанных распределений выполнялся независимо. В случае формирования статистики по каждому атакующему хосту, классовые метки подсчитывались по каждому хосту (AH_1 … AH_5). В случае формирования статистики по каждому типу КА – подсчитывались по каждому типу КА («Отказ в обслуживании»; «Сетевая разведка»; «Фаззинг»).

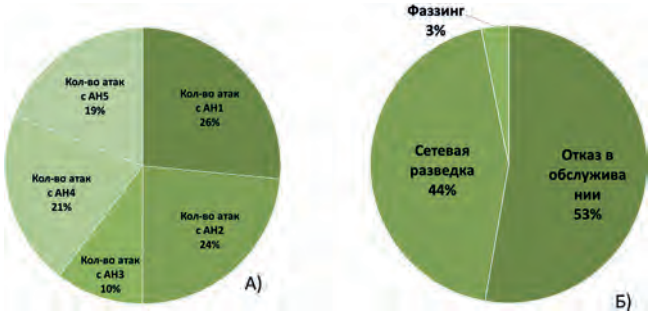

Рис.6. Распределение КА:
а) – с каждого атакующего хоста; б) –каждого типа.

Всего совокупно с каждого хоста было собрано 488 120 ед. записей о проведении КА. Отметим, что суммарное количество классовых меток превышает объем собранных данных вследствие наличия многозначности в данных. Дополнительно отметим также, что из-за независимого подсчета частотной статистики для распределения атак с **каждого атакующего хоста** и распределение атак каждого **типа**, вследствие различной размерности пространства классовых меток и **наличия многозначных записей**, суммарный объем классовых меток различается. Различия связаны с одновременно реализуемыми КА одного типа по разным хостам.

На рис. 7.а приведено распределение классовых меток **по количеству хостов**, одновременно участвующих в КА, а на рис. 7.б распределение классовых меток по каждому **типу** КА, одновременно участвующих в атаке на VH_1. В отличии от рис. 6, классовые метки подсчитывались по каждому хосту (AH_1 … AH_5).

Количество классовых меток, связанных с отсутствием КА, составляет. 20% от общего количества классовых меток (51605 ед).

Анализ гистограмм, характеризующих информационную значимость атрибутов КС, показал, что для некоторых атрибутов многозначной КА их значимость

Рис. 7. Распределение классовых меток:
а) По одновременно задействованным хостам при реализации КА;
б) По одновременно совершаемым типам КА.

превышает информационную значимость однозначных КА, входящих в ее состав. Обнаружено, что многозначная КА (совокупность однозначных КА) является отдельной сущностью, обладающей собственным распределением информативной значимости атрибутного пространства.

Оценим и сравненим информативность атрибутов, собранных в результате эксперимента с использованием ПАК. Оценка значимости атрибутного пространства производилась с использованием библиотеки scikit-learn; модуля SelectKBest. Анализ доступных функций определил наиболее подходящую для решения поставленной задачи – chi2. Дополнительно для оценки информативности атрибутов использовался индекс Джини, вычисляемый с использованием реализации алгоритма Random Forest (RF) библиотекой scikit-learn

На рис. 8 приведена информационная значимость каждого из 115 атрибутов для 23 комбинаций КА. Рис. 8.а соответствует анализу информативности каждого атрибута по критерию chi2; рис. 8.б – анализу информативности каждого атрибута по индексу Джини (RF).

Анализ распределения гистограмм позволяет сделать два вывода. Каждый алгоритм оценки информативности формирует уникальную «картину» распределений значимости атрибутов. Например, алгоритм оценки информативности посредством по критерию chi2, выделил группу атрибутов, связанную с частотой центрального процессора, как значимую для многозначных атак, вызывающих «отказ в обслуживании». При этом индекс Джини для этих атрибутов незначителен. Аналогичная ситуация для распределения

Рис.8. Информационная значимость каждого из 115 атрибутов для 23 комбинаций КА:
а) анализ информативности каждого атрибута по критерию chi2;
б) анализ информативности каждого атрибута по индексу Джини (RF).

значимости атрибутов по критерию «индекс Джини» (RF). Вторым выводом является то, что разные КА имеют разное распределение значимых атрибутов. Анализ «спектра» полученных распределений выявил разделимость КА по информативности атрибутов, что позволяет решать задачу классификации наблюдаемых атак на имеющихся данных. Анализ гистограмм показывает, что многозначная КА, хоть и сочетает в себе признаки однозначных КА, являющихся ее составляющими, но является отдельной сущностью, чье распределение невозможно получить посредством сложения распределений однозначных КА.

В результате, как показано в [17], работа с многозначной КА как с отдельной сущностью может повысить точность классификации и прогнозирования, что обусловлено как особенностями алгоритмов МО, работающих в режиме обучения «с учителем», так и синергетическим эффектом многозначных данных.

Как известно, под синергией понимается усиливающий эффект взаимодействия двух или более факторов, характеризующийся тем, что совместное действие этих факторов существенно превосходит простую сумму действий каждого из указанных факторов. Так, например, в [17] алгоритмы МО на базе искусственной нейронной сети, работающие в режиме обучения «с учителем», чья архитектура предполагает наличие «скрытых» слоев и состояний, способны к выделению записей с многозначной классовой меткой (КА) в отдельный кластер. Выделение записей производится за счет формирования отдельных решающих правил для таких записей посредством выделения посредством корректировки ряда весовых коэффициентов (в случае искусственной нейронной сети; отдельных решающих деревьев в случае древовидных алгоритмов).

Поскольку многозначная КА является отдельной сущностью, эта сущность может быть выявлена алгоритмами МО с высокой обобщающей способностью, способными к кластеризации (например, искусственная нейронная сеть с включением самоорганизующихся карт Кохонена [18,19] в свою структуру). Если алгоритм МО не подразумевает многозначный выход, то даже при правильно выделенном кластере «внутри», отсутствие многозначного выхода приводит к ошибке классификации по двум причинам. Первой причиной является несовпадение результирующей однозначной классовой метки и многозначной эталонной метки (если данные размечены как многозначные) [20,21]. Второй причиной является «наложения» решающих правил, связанных с отнесением неразмеченной записи либо к многозначной КА, либо к однозначной КА, входящей в состав многозначной [22]. В случае некорректной разметки исходных данных (данные маркируются как однозначные при наличии многозначных записей), ошибка классификации (прогнозирования) происходит из-за несовпадения результирующей многозначной классовой метки и однозначной эталонной метки.

Выводы

Описывается создание ПАК для сбора телеметрии в ходе имитационного моделирования КА в КС, обладающих свойством *многозначности* в табличном представлении. Данные, порождаемые ПАК, могут быть использованы для противодействия КА при разработке СОВ, учитывающих многозначность «исторических» записей и интерпретироваться как средство обнаружения КА в КС.

ПАК имитирует реальные данные, соответствующие задачам информационной безопасности. За счет большого количества настраиваемых параметров моделирования КА, возможна «тонкая» настройка распределения классовых меток и соотношения доли однозначных и многозначных записей в данных, формируемых ПАК.

Новизна разработанного ПАК заключается автоматизированной параллельной маркировке всех КА, осуществляемых на КС, что позволяет учесть многозначность уже на этапе сбора данных.

С использованием разработанного ПАК, сформирован многозначный набор данных, представляющий собой диагностическую информацию о сети, подвергаемой 3 типам КА, совершаемым параллельно – «Отказ в обслуживании»; «Сетевая разведка»; «Фаззинг». В рамках реализации ККА типа «отказ в обслуживании» проведены атаки по по протоколам *ICMP*, *UDP* и *TCP*; в рамках реализации ККА типа «Сетевая разведка» проводились КА «сканирование портов» и «сканирование операционной системы».

Реализован новый фреймворк для решения задач классификации, прогнозирования и исследования гиперпараметров ИНС с множественным выходом для многозначных данных.

Обнаружено, что многозначная КА (совокупность однозначных КА) является отдельной сущностью с собственным распределением информативной значимости атрибутного пространства. Поскольку многозначная КА является отдельной сущностью, эта сущность может быть выявлена алгоритмами МО с высокой обобщающей способностью, способными к кластеризации. Если алгоритм МО не подразумевает многозначный выход, то даже при правильно выделенном кластере «внутри», отсутствие многозначного выхода приводит к ошибке классификации. Работа с многозначной КА как с отдельной сущностью, повышает точность классификации и прогнозирования.

Литература

1. *Котенко И. В., Дун Х. Обнаружение атак в интернете вещей на основе многозадачного обучения и гибридных методов сэмплирования // Вопросы кибербезопасности. 2024. Т. 60, № 2. С. 10–21. DOI: 10.21681/2311-3456-2024-2-10-21.*
2. *Рзаев Б. Т., Лебедев И. С. Применение бэггинга при поиске аномалий сетевого трафика // Научно-технический вестник информационных технологий, механики и оптики. 2021. Т. 21, № 2. С. 234–240. DOI: 10.17586/2226-1494-2021-21-2-234-240.*
3. *Solomin A. A., Ivanova Yu. A. Modern approaches to multiclass intent classification based on pre-trained transformers // Naučno-teh. vestn. inf. tehnol. meh. opt. 2020. Т. 20, № 4. С. 532–538. DOI: 10.17586/2226-1494-2020-20-4-532-538.*
4. *Лебедев И. В., Симонян А. Г. Анализ Трафика Для Исследования Сетевой Активности И Обнаружения Атак // Сборник трудов XIV Международной отраслевой научно-технической конференции. 2020. Москва: ООО «Издательский дом Медиа паблишер», 2020. С. 215–216.*
5. *Du Z., He K., Lui W., He W. Automated Neural Machine Translation for Icd Coding // Industry and agriculture. Т. 66, № 1. С. 41–58.*
6. *Бергер А. И., Гуда С. А. Свойства алгоритмов поиска оптимальных порогов для задач многозначной классификации // Компьютерные исследования и моделирование. 2022. Т. 14, № 6. С. 1221–1238.*
7. *Karpovich S. N. Multi-Label Classification of Text Documents using Probabilistic Topic Modeling // SPIIRAS Proceedings. 2016. Т. 4, № 47. С. 92–104. DOI: 10.15622/sp.47.5.*
8. *Раковский Д. И. Влияние проблемы многозначности меток классов системных журналов на защищенность компьютерных сетей // Наукоемкие Технологии В Космических Исследованиях Земли. 2023. Т. 15, № 1. С. 48–56с. DOI: 10.36724/2409-5419-2023-15-1-48-56.*
9. *Talukder Md. A., Hasan K. F., Islam Md. M., Uddin Md. A., Akhter A., Yousuf M. A., Alharbi F., Moni M. A. A dependable hybrid machine learning model for network intrusion detection // Journal of Information Security and Applications. 2023. Т. 72. С. 103405. DOI: 10.1016/j.jisa.2022.103405.*
10. *Riera T. S., Higuera J. -R. B., Higuera J. B., Herraiz J. -J. M., Montalvo J. -A. S. A new multi-label dataset for Web attacks CAPEC classification using machine learning techniques // Computers & Security. 2022. Т. 120. С. 102788. DOI: 10.1016/j.cose.2022.102788.*
11. *Кондаков С. Е., Рудь И. С. Модель процесса проведения компьютерных атак с использованием специальных информационных воздействий // Вопросы Кибербезопасности. 2021. Т. 45, № 5. С. 12–20. DOI: 10.21681/2311-3456-2021-5-12-20.*
12. *Шелухин О. И., Раковский Д. И. Визуализация Аномальных Событий При Прогнозировании Состояний Компьютерных Систем На Основе «Исторических Данных» // Reds: Телекоммуникационные Устройства И Системы. 2022. Т. 12, № 2. С. 53–58.*
13. *Vasilyev V., Kirillova A., Vulfin A., Nikonov A. Cybersecurity Risk Assessment Based on Cognitive Attack Vector Modeling with CVSS Score // 2021 International Conference on Information Technology and Nanotechnology (ITNT). Samara, Russian Federation: IEEE, 2021. С. 1–6. DOI: 10.1109/ITNT52450.2021.9649191.*
14. *Раковский Д. И. Обнаружение компьютерных атак и предупреждение нарушений функционирования компьютерных сетей на основе многозначных закономерностей // Сборник трудов III Всероссийской научной школы-семинара «Современные тенденции развития методов и технологий защиты информации». 2023. С. 307–311.*
15. *Riera T. S., Higuera J. -R. B., Higuera J. B., Herraiz J. -J. M., Montalvo J. -A. S. A new multi-label dataset for Web attacks CAPEC classification using machine learning techniques // Computers & Security. 2022. Т. 120. С. 102788. DOI: 10.1016/j.cose.2022.102788.*
16. *Кажемский М. А., Шелухин О. И. Многоклассовая Классификация Сетевых Атак На Информационные Ресурсы Методами Машинного Обучения // Труды Учебных Заведений Связи. 2019. Т. 5, № 1. С. 107–115. DOI: 10.31854/1813-324X-2019-5-1-107-115.*
17. *Шелухин О. И., Раковский Д. И. Многозначная классификация компьютерных атак с использованием искусственных нейронных сетей с множественным выходом // Труды учебных заведений связи. 2023 Т. 9, № 4. С. 95–111. DOI:10.31854/1813-324X-2023-9-4-97-113*
18. *Kukartsev V., Nelyub V., Kozlova A., Borodulin A., Rukosueva A. Intelligent Data Analysis as a Method of Determining the Influence of Various Factors on the Level of Customer Satisfaction of the Company // Data Analytics in System Engineering / под ред. Silhavy R., Silhavy P. Cham: Springer Nature Switzerland, 2024. Т. 935. С. 109–128. DOI: 10.1007/978-3-031-54820-8_11.*
19. *Karnaukh S. G., Markov O. E., Kukhar V. V., Shapoval A. A. Classification of steels according to their sensitivity to fracture using a synergetic model // Int J Adv Manuf Technol. 2022. Т. 119, № 7–8. С. 5277–5287. DOI: 10.1007/s00170-022-08653-y.*
20. *Zhang X., Zhuang Y., Zhang T., Li C., Chen H. Masked Image Modeling Auxiliary Pseudo-Label Propagation with a Clustering Central Rectification Strategy for Cross-Scene Classification // Remote Sensing. 2024. Т. 16, № 11. С. 1983. DOI: 10.3390/rs16111983.*
21. *Zhao T., Zhang Y., Miao D., Zhang H. Multi-granular labels with three-way decisions for multi-label classification // Int. J. Mach. Learn. & Cyber. 2023. Т. 14, № 11. С. 3737–3752. DOI: 10.1007/s13042-023-01861-2.*
22. *Priyadharshini M., Banu A. F., Sharma B., Chowdhury S., Rabie K., Shongwe T. Hybrid Multi-Label Classification Model for Medical Applications Based on Adaptive Synthetic Data and Ensemble Learning // Sensors. 2023. Т. 23, № 15. С. 6836. DOI: 10.3390/s23156836.*

References

1. *Kotenko I. V., Dun H. Obnaruzhenie atak v internete veshhej na osnove mnogozadachnogo obuchenija i gibridnyh metodov sjemplirovanija // Voprosy kiberbezopasnosti. 2024. Т. 60, № 2. S. 10–21. DOI: 10.21681/2311-3456-2024-2-10-21.*
2. *Rzaev B. T., Lebedev I. S. Primenenie bjegginga pri poiske anomalij setevogo trafika // Nauchno-tehnicheskij vestnik informacionnyh tehnologij, mehaniki i optiki. 2021. Т. 21, № 2. S. 234–240. DOI: 10.17586/2226-1494-2021-21-2-234-240.*
3. *Solomin A. A., Ivanova Yu. A. Modern approaches to multiclass intent classification based on pre-trained transformers // Naučno-teh. vestn. inf. tehnol. meh. opt. 2020. Т. 20, № 4. S. 532–538. DOI: 10.17586/2226-1494-2020-20-4-532-538.*
4. *Lebedev I. V., Simonjan A. G. Analiz Trafika Dlja Issledovanija Setevoj Aktivnosti I Obnaruzhenija Atak // Sbornik trudov XIV Mezhdunarodnoj otraslevoj nauchno-tehnicheskoj konferencii. 2020. Moskva: OOO «Izdatel'skij dom Media pablisher», 2020. S. 215–216.*
5. *Du Z., He K., Lui W., He W. Automated Neural Machine Translation for Icd Coding // Industry and agriculture. Т. 66, № 1. S. 41–58.*
6. *Berger A. I., Guda S. A. Svojstva algoritmov poiska optimal'nyh porogov dlja zadach mnogoznachnoj klassifikacii // Komp'juternye issledovanija i modelirovanie. 2022. Т. 14, № 6. S. 1221–1238.*

7. *Karpovich S. N. Multi-Label Classification of Text Documents using Probabilistic Topic Modeling // SPIIRAS Proceedings. 2016. T. 4, № 47. S. 92–104. DOI: 10.15622/sp.47.5.*
8. *Rakovskij D. I. Vlijanie problemy mnogoznachnosti metok klassov sistemnyh zhurnalov na zashhishhennost' komp'juternyh setej // Naukoemkie Tehnologii V Kosmicheskih Issledovanijah Zemli. 2023. T. 15, № 1. S. 48–56s. DOI: 10.36724/2409-5419-2023-15-1-48-56.*
9. *Talukder Md. A., Hasan K. F., Islam Md. M., Uddin Md. A., Akhter A., Yousuf M. A., Alharbi F., Moni M. A. A dependable hybrid machine learning model for network intrusion detection // Journal of Information Security and Applications. 2023. T. 72. S. 103405. DOI: 10.1016/j.jisa.2022.103405.*
10. *Riera T. S., Higuera J. -R. B., Higuera J. B., Herraiz J. -J. M., Montalvo J. -A. S. A new multi-label dataset for Web attacks CAPEC classification using machine learning techniques // Computers & Security. 2022. T. 120. S. 102788. DOI: 10.1016/j.cose.2022.102788.*
11. *Kondakov S. E., Rud' I. S. Model' processa provedenija komp'juternyh atak s ispol'zovaniem special'nyh informacionnyh vozdejstvij // Voprosy Kiberbezopasnosti. 2021. T. 45, № 5. S. 12–20. DOI: 10.21681/2311-3456-2021-5-12-20.*
12. *Sheluhin O. I., Rakovskij D. I. Vizualizacija Anomal'nyh Sobytij Pri Prognozirovanii Sostojanij Komp'juternyh Sistem Na Osnove «Istoricheskih Dannyh» // Reds: Telekommunikacionnye Ustrojstva I Sistemy. 2022. T. 12, № 2. S. 53–58.*
13. *Vasilyev V., Kirillova A., Vulfin A., Nikonov A. Cybersecurity Risk Assessment Based on Cognitive Attack Vector Modeling with CVSS Score // 2021 International Conference on Information Technology and Nanotechnology (ITNT). Samara, Russian Federation: IEEE, 2021. S. 1–6. DOI: 10.1109/ITNT52450.2021.9649191.*
14. *Rakovskij D. I. Obnaruzhenie komp'juternyh atak i preduprezhdenie narushenij funkcionirovanija komp'juternyh setej na osnove mnogoznachnyh zakonomernostej // Sbornik trudov III Vserossijskoj nauchnoj shkoly-seminara «Sovremennye tendencii razvitija metodov i tehnologij zashhity informacii». 2023. S. 307–311.*
15. *Riera T. S., Higuera J. -R. B., Higuera J. B., Herraiz J. -J. M., Montalvo J. -A. S. A new multi-label dataset for Web attacks CAPEC classification using machine learning techniques // Computers & Security. 2022. T. 120. S. 102788. DOI: 10.1016/j.cose.2022.102788.*
16. *Kazhemskij M. A., Sheluhin O. I. Mnogoklassovaja Klassifikacija Setevyh Atak Na Informacionnye Resursy Metodami Mashinnogo Obuchenija // Trudy Uchebnyh Zavedenij Svjazi. 2019. T. 5, № 1. S. 107–115. DOI: 10.31854/1813-324X-2019-5-1-107-115.*
17. *Sheluhin O. I., Rakovskij D. I. Mnogoznachnaja klassifikacija komp'juternyh atak s ispol'zovaniem iskusstvennyh nejronnyh setej s mnozhestvennym vyhodom // Trudy uchebnyh zavedenij svjazi. 2023 T. 9, № 4. S. 95–111. DOI:10.31854/1813-324X-2023-9-4-97-113*
18. *Kukartsev V., Nelyub V., Kozlova A., Borodulin A., Rukosueva A. Intelligent Data Analysis as a Method of Determining the Influence of Various Factors on the Level of Customer Satisfaction of the Company // Data Analytics in System Engineering / pod red. Silhavy R., Silhavy P. Cham: Springer Nature Switzerland, 2024. T. 935. S. 109–128. DOI: 10.1007/978-3-031-54820-8_11.*
19. *Karnaukh S. G., Markov O. E., Kukhar V. V., Shapoval A. A. Classification of steels according to their sensitivity to fracture using a synergetic model // Int J Adv Manuf Technol. 2022. T. 119, № 7–8. S. 5277–5287. DOI: 10.1007/s00170-022-08653-y.*
20. *Zhang X., Zhuang Y., Zhang T., Li C., Chen H. Masked Image Modeling Auxiliary Pseudo-Label Propagation with a Clustering Central Rectification Strategy for Cross-Scene Classification // Remote Sensing. 2024. T. 16, № 11. S. 1983. DOI: 10.3390/rs16111983.*
21. *Zhao T., Zhang Y., Miao D., Zhang H. Multi-granular labels with three-way decisions for multi-label classification // Int. J. Mach. Learn. & Cyber. 2023. T. 14, № 11. S. 3737–3752. DOI: 10.1007/s13042-023-01861-2.*
22. *Priyadharshini M., Banu A.F., Sharma B., Chowdhury S., Rabie K., Shongwe T. Hybrid Multi-Label Classification Model for Medical Applications Based on Adaptive Synthetic Data and Ensemble Learning // Sensors. 2023. T. 23, № 15. S. 6836. DOI: 10.3390/s23156836.*