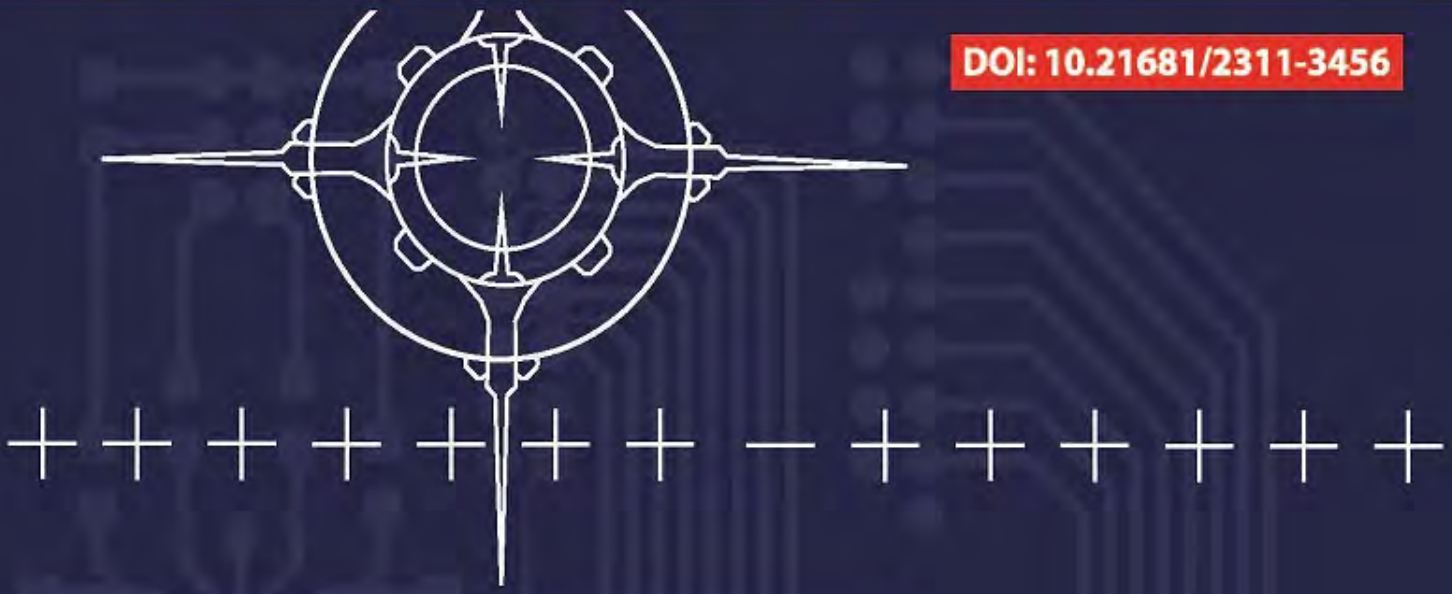
ВОПРОСЫ КИБЕРБЕЗОПАСНОСТИ

№3 2024 (61)

DOI: 10.21681/2311-3456



Концептуальные вопросы кибербезопасности

Безопасный искусственный интеллект

Аудит информационной безопасности

Эшелон
комплексная безопасность

{KOMRAD}
Enterprise SIEM

ВЫСОКАЯ ПРОИЗВОДИТЕЛЬНОСТЬ И МИНИМАЛЬНЫЕ ТРЕБОВАНИЯ К АППАРАТНОМУ ОБЕСПЕЧЕНИЮ

KOMRAD Enterprise SIEM позволяет осуществлять централизованный сбор событий ИБ, выявлять инциденты ИБ и оперативно на них реагировать. Применение комплекса позволяет эффективно выполнять требования, предъявляемые регуляторами к защите персональных данных, к обеспечению безопасности государственных информационных систем и контролю критической информационной инфраструктуры предприятия. KOMRAD позволяет отправлять данные о событиях и инцидентах ИБ во внешние системы (например, ГосСОПКА).

- Визуальный конструктор запросов и директив корреляции
- Высокая производительность
- Гибкая интеграция с нестандартными источкниками событий
- Широкий спектр поддержки источников событий
- Ролевая модель управления доступом
- Оперативное оповещение об инциденте
- Масштабируемость

Чтобы получить демо-версию KOMRAD Enterprise SIEM или заказать пилот у наших партнеров в вашем регионе, свяжитесь с нашим отделом продаж по e-mail: sales@npo-echelon.ru.

ВОПРОСЫ КИБЕРБЕЗОПАСНОСТИ

НАУЧНЫЙ РЕЦЕНЗИРУЕМЫЙ ЖУРНАЛ

№3 (61) 2024 г.

Выходит 6 раз в год

Журнал выходит с 2013 г. (Свидетельство о регистрации ПИ № ФС77-75239). Перерегистрировано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций 07.03.2019.

Журнал входит в рейтинг научных изданий ВАК в категории К1, индексируется в RSCI, публикует статьи по специальностям 1.2.4 и 2.3.6 - физ-мат.науки; 2.2.15, 2.3.1, 2.3.5, 2.3.6 -техн.науки

Учредитель и издатель

АО «Научно-производственное объединение «Эшелон»

Над номером работали:

Г. И. Макаренко – *шеф-редактор*, И. Ю. Шахалов – *отв. секретарь*, Т. В. Галатонов – *сайт*, Н. С. Рождественская – *маркетинг и подписка*

Подписано к печати 15.05.2024 г.
Общий тираж 120 экз. Цена свободная

Адрес: 107023, Москва, ул. Электрозаводская, д. 24, стр. 1.
E-mail: editor@cyberrus.info, тел.: +7 (985) 939-75-01.
Требования, предъявляемые к рукописям, размещены на сайте: **https://cyberrus.info/**

СОДЕРЖАНИЕ

Подписка на журнал осуществляется в почтовых отделениях по каталогу «Пресса России». Подписной индекс 40707

КОМБИНИРОВАНИЕ СПОСОБОВ ВЫЯВЛЕНИЯ ИНСАЙДЕРОВ БОЛЬШИХ ИНФОРМАЦИОННЫХ СИСТЕМ

Буйневич М. В.[1], Власов Д. С.[2], Моисеенко Г. Ю.[3]

DOI: 10.21681/2311-3456-2024-3-2-13

Цель исследования: *изыскание направлений повышения эффективности противодействия инсайдерам в больших информационных системах за счет комбинирования способов их выявления.*

Методы исследования: *аналитический обзор релевантных научных публикаций, концептуальное моделирование, формализация, категориальный подход, экспертное и теоретическое комбинирование, синтез, алгоритмизация.*

Полученные результаты: *получен обобщенный список и разработана частично формализованная модель объединения качественно различных способов выявления инсайдеров в больших информационных системах; предложен экспертный прогноз 21 комбинации из 7 указанных способов, дана теоретическая оценка успешности их сочетания; синтезирован комбинированный способ выявления инсайдеров, алгоритм которого задан в виде псевдокода.*

Научная новизна работы *определяется авторским подходом к комбинированию способов на основе категориального пространства, которое имеет оси вдоль следующих пар антагонистических элементов: нормальное vs аномальное, статическое vs динамическое, субъект vs объект. Большинство комбинаций способов предложены впервые.*

Ключевые слова: *большие информационные системы, информационная безопасность, инсайдер, способ выявления, комбинация методов.*

METHODS COMBINING FOR IDENTIFYING OF INSIDERS IN LARGE INFORMATION SYSTEMS

Buinevich M. V.[4], Vlasov D. S.[5], Moiseenko G. Y.[6]

The goal of the investigation: *finding ways to improve the effectiveness of countering insiders in large information systems by combining methods of their detection.*

Research methods: *analytical review of relevant scientific publications, conceptual modeling, formalization, categorical approach, expert and theoretical combination, synthesis, algorithmization.*

Results: *a generalized list is obtained and a partially formalized model of combining qualitatively different methods of detecting insiders in large information systems is developed; an expert forecast of 21 combinations from 7 of these methods is proposed, a theoretical evaluation of the success of their combination is given; a combined method of detecting insiders is synthesized, the algorithm of which is given in the form of pseudo code.*

The scientific novelty *is determined by the author's approach to combining methods on the basis of a categorical space with axes along the following pairs of antagonistic elements: normal vs. abnormal, static vs. dynamic, subject vs. object. Most of the combinations of methods are proposed for the first time.*

Keywords: *large information system, information security, insider, detection method, combination of methods.*

1 Буйневич Михаил Викторович, доктор технических наук, профессор, профессор кафедры прикладной математики и информационных технологий Санкт-Петербургского университета государственной противопожарной службы МЧС России, Санкт-Петербург. ORCHID: https://orcid.org/0000-0001-8146-0022. Scopus Author ID: 56122749800. E-mail: bmv1958@yandex.ru

2 Власов Дмитрий Сергеевич, начальник управления информационных технологий и связи Главного управления МЧС России по г. Санкт-Петербургу, Россия. ORCHID: http://orcid.org/0000-0003-2332-8431. E mail: prikerx@bk.ru

3 Моисеенко Григорий Юрьевич, руководитель направления Министерства обороны РФ, Москва, Россия. E-mail: mogreq@mail.ru

4 Mikhail V. Buinevich, Dr.Sc., Professor, Professor of Dep. Applied Mathematics and Information Technologies of Saint-Petersburg University of State Fire Service of EMERCOM of Russia, Saint-Petersburg, Russia. ORCHID: https://orcid.org/0000-0001-8146-0022. Scopus Author ID: 56122749800. E-mail: bmv1958@yandex.ru

5 Dmitry S. Vlasov, Head of Information Technology and Communications Department EMERCOM of Russia Main Directorate in the St. Petersburg city, Saint-Petersburg, Russia. ORCHID: http://orcid.org/0000-0003-2332-8431. E-mail: prikerx@bk.ru

6 Grigory Y. Moiseenko, Head of direction, Ministry of Defense of the Russian Federation, Moscow, Russia. E-mail: mogreq@mail.ru

Введение

Согласно отчету «2023 Cost of Insider Threats: Global Report» [1], опубликованному Ponemon Institute, специализирующемуся на независимых исследованиях методов обеспечения конфиденциальности в сфере бизнеса и государственного управления, киберугрозы со стороны инсайдеров выросли за последний год на 47%. За тот же период расходы на раннее обнаружение инсайдеров и нейтрализацию их атак возросли на 31%. Отчет «Data Breach Investigations Report 2023» от компании Verizon [2], составленный на основе обработки свыше 16 000 зарегистрированных инцидентов безопасности и более 5000 подтвержденных случаев утечки данных на 6 континентах и в 20 отраслях, показал, что около трети всех случаев произошло вследствие инсайдерских атак.

Такая тенденция, в том числе, свидетельствует, что применяемые способы противодействия инсайдерской угрозе достигли некого предела своей эффективности. Ситуация усложняется следующими обстоятельствами. Во-первых, не все инсайдеры и «среды их обитания» одинаковы. Такие нарушители существенно различаются по мотивации (халатные сотрудники, злоумышленники, агенты влияния и т.д.) [3], поведению и подготовке, а в разных отраслях доминируют собственные инсайдерский атаки (например, здравоохранение – социальная инженерия, ИТ-сектор – ошибки привилегированных пользователей, сектор финансовых услуг – умышленная кража учетных записей). Отсюда следует, что конкретный способ выявления инсайдеров может быть максимально результативным только для сотрудников или компаний (организаций) определенного типа. Во-вторых, для инсайдерской угрозы время играет важную роль. В исследовании [2] указывается, что среднее время нейтрализации инсайдера составляет 77 дней, и только 13% инсайдеров нейтрализуются быстрее, чем за 30 дней. Кроме того, результаты исследования доказывают, что инциденты продолжительностью свыше 90 дней, обходятся компаниям примерно в 13,7 млн. долларов США в год, а менее 30 дней – примерно в 2 раза меньше. Соответственно, способ выявления инсайдеров должен быть максимально оперативным. И, в-третьих, на успешность атак оказывает влияние размер самой компании. Вполне естественно, что крупные организации, имеющие большие информационные системы, подвергаются большему количеству инсайдерских атак более многообразной этимологии (т.е. происхождения или природы). По данным [1] ежегодные расходы крупных организаций, владельцев больших информационных систем, на сдерживание инсайдеров составили в среднем 18,3 млн. долларов США против 7,68 млн. долларов США у компаний численностью менее 500 сотрудников. Поэтому способ выявления инсайдеров должен быть возможно более экономичным.

С учетом сложившихся обстоятельств рациональным решением для больших информационных систем представляется комбинирование способов выявления инсайдеров. И первыми шагами в этом направлении является их попарное сочетание с доказательством реализуемости комбинации через формализацию и последующий мыслительный эксперимент.

Обзор релевантных работ

Произведем обзор (в том числе и авторских) работ, в которых описываются подходы (или способы), применимые для комбинирования различных методов, моделей и инструментов выявления инсайдеров.

Работа [4] посвящена выявлению инсайдеров путем анализа сетевой активности сотрудников организации, основываясь на двух подходах к классификации каждого пользователя, как лояльного или злонамеренного. Суть первого заключается в применении строгих правил, созданных экспертом (или их группой); суть второго – в создании моделей машинного обучения по выборке сетевого трафика, полученной с помощью генератора сетевых атак. В качестве классификаторов машинного обучения применялись такие базовые как деревья решений, наивный байесовский классификатор, метод k-ближайших соседей и метод опорных векторов, а также их комбинация: голосование большинством, взвешенное и мягкое голосование, Adaboost. Поскольку каждый из подходов мог давать собственные результаты классификации пользователей, в т.ч. дополняющие или противоречащие друг другу, то для их комбинирования применялись 4 вариации: объединение, пересечение или выбор одного из вышеперечисленных.

В работе [5] описано решение задачи противодействия атакам на сервисы облачных вычислений со стороны инсайдеров, для чего комбинируется пара методов путем их наложения. Первый основан на классическом дереве атак, примененном к внутренней среде облачных сервисов; второй представляет собой, так называемую, цепочку уничтожения, вышедшую из концепции ведения боевых действий и заключающуюся в отслеживании степени продвижения атаки к заданным целям. Как результат, появляется возможность оценивать защищаемую систему на различных уровнях абстракции.

Работа [6] ссылается на 2 метода противодействия инсайдерам: на основе анализа внутренних угроз в организации и генерации данных, соответствующих человеческому поведению, его психологическим аспектам и контрразведывательной

деятельности. Как результат, второй метод дополняет первый, делая его более точным при фактическом обнаружении инсайдеров.

В [7] описывается способ, где сначала производится сбор данных с устройств сотрудников (телефоны, ноутбуки, персональные компьютеры и пр.) для создания профиля каждого из пользователей. А затем, с применением методов машинного обучения в части поиска аномалий выделяются те профили, которые имеют существенное отличие от профилей других сотрудников. Идея, предлагаемая в статье, основана на том, что инсайдеры будут иметь «концентрацию» данных, отличную от тех, которая есть на устройствах у коллег. Таким образом, предлагаемая система основана на объединении инструментов сбора и обработки данных, которые были подвергнуты определенному упрощению. Как результат, удается обнаруживать инсайдеров, профили которых отличаются не только от профилей других коллег, но и от собственных, но на более раннем периоде. Второй случай, очевидно, означает факт превращения лояльного сотрудника в нарушителя, что позволяет лучше понять причины такого превращения и негативные предпосылки этого для организации.

Работа [8] аналогична [7] в части подхода к выявлению инсайдерской деятельности – через выявление сотрудников, поведение которых имеет существенные отличия от остальных. Однако в исследовании подчеркивается, что современные инсайдеры стремятся быть похожими на лояльных пользователей. Таким образом, для их выявления может потребоваться анализ нескольких источников информации, в интересах чего авторы предлагают комбинировать консенсусную кластеризацию (позволяющую оценить влияние небольших возмущений в наборах данных на состав кластера) и обнаружение аномалий. Основная идея заключается в поиске аномальных признаков типовой активности сотрудников, когда действия инсайдера внешне неотличимы от действий лояльных пользователей; это достигается каскадным применением методов машинного обучения.

В работе [9] предлагается способ предупреждения инсайдерских атак, объединяющий сразу 3 метода риск-менеджмента – основанный на поведении злоумышленника и учитывающие компьютерные и психо-социальные риски. В первом методе выделяются такие риски, как недовольный сотрудник, принятие критики, управление гневом, невовлеченность в «жизнь» организации, игнорирование правил, производительность, стресс, конфронтация, личные проблемы, эгоцентризм, надежность, прогулы. Ко второму методу относятся следующие риски: неудачный вход в систему, подозрительное общение, сбор данных, установка и использование «нештатного» программного обеспечения, удаленный вход, несанкционированный доступ, удаление логов. Для третьего метода характерны такие риски, как проблемы с деньгами, недавний разрыв или потеря, чрезмерная депрессия, патологическое отыгрывание (игромания), расстройство адаптации (т.е. чрезмерное реагирование на стресс) и проблемы с тревогой (т.е. бессознательное развитие тревожной для человека ситуации). Для моделирования инсайдерской деятельности используется системная динамика (направление в изучении сложных систем). Как результат, с применением продукта Vensim строится модель, связывающая вероятностное поведение человека и детерминированное поведение системы.

В работе [10] описываются две модели, работающие по качественно разным признакам выявления инсайдеров. Первая предназначена для распознавания лица пользователя и сравнение его с занесенным в базу данных при регистрации, для чего используется OpenCV (*аббр. от англ.* Open Source Computer Vision, *пер. на русск.* открытая библиотека для работы с алгоритмами компьютерного зрения). Вторая отслеживает поведение пользователей и классифицирует их на 4 следующих: легитимный, возможно легитимный, возможно нелегитимный, нелегитимный. Эта модель реализуется на базе алгоритма k-ближайших соседей (из области машинного обучения). Обе модели объединяются в одну метамодель, которая собственно и лежит в основе способа обнаружения инсайдеров.

Анализ результатов обзора релевантных работ позволяет сделать следующие выводы. Во-первых, существует достаточно малое количество исследований, направленных на объединение разнородных способов выявления инсайдеров. Во-вторых, наблюдается тенденция применения машинного обучения [7, 8, 10], хотя оно и подходит только для определенных способов поиска, имеющих возможность получения формализованной Best Practices. И, в-третьих, объединение способов, относящихся к разным областям организаций (например, сетевое поведение пользователей и их психоэмоциональное состояние) является существенной проблемой, не нашедшей полноценного решения; частично решения предлагаются в [6, 10]. Можно отметить работу [9], где подобная попытка предпринята путем применения системно-динамического моделирования. Однако какого-либо полноценного подхода для объединения всех способов или их большей части на данный момент не обнаружено.

Способы выявления инсайдеров

Систематизируем и обобщим способы выявления инсайдеров в больших информационных системах с учетом их списков, составленных авторами ранее [11, 12].

Способ 1 – Анализ динамики обычной жизни, основанный на учете событий и ситуаций сотрудников, в том числе и вне организации. Так, повышение финансовых трат и большое количество взятых кредитов может привести к тому, что сотрудник станет продавать конфиденциальную информацию «третьим лицам».

Способ 2 – Выявление аномалий в типовых сценариях работы пользователей, основанное на модели IDES (*аббр. от англ.* Intrusion Detection Expert System, *пер. на русск.* экспертная система обнаружения вторжений) и подразумевающее некоторые отклонения в действиях потенциального нарушителя по сравнению с поведением большинства других – лояльных. Например, резкое повышение отправленного сетевого трафика от сотрудника на внешний Интернет-ресурс может сигнализировать о потенциальной угрозе нарушения конфиденциальности информации.

Способ 3 – Предотвращение накопления критической конфиденциальной информации, заключающееся в отслеживании объема и/или охвата данных, к которым получил доступ сотрудник. Так, если кем-либо, в нарушение должностных обязанностей и превышение полномочий, в организации была собрана по частям достаточно полная клиентская база, то это может считаться подозрительным и сигнализировать о целенаправленной инсайдерской деятельности.

Способ 4 – «Ловля на живца» (*в англ. лит.* – Honeypot, Honeynet, Honeytoken и пр., *досл. пер. на русск.* «медовая ловушка»), суть которой заключается в оставлении «приманок» для злоумышленника, которые не имеют особой ценности и существенной защиты, но получение доступа к которым будет говорить о злонамеренной деятельности. Например, если к хранимому в условно свободном доступе документу с лже-финансовой отчетностью будут осуществляться попытки доступа (с целью хищения или уничтожения), то это может сигнализировать о заинтересованности сотрудника-инсайдера в неправомерных действиях.

Способ 5 – Выявление потенциального нарушителя психодиагностическими методами, которое основано на изучении психологии сотрудников и определения среди них тех, кто является или может стать инсайдером. Так, сотрудники с высоким желанием самоутвердиться, придерживающиеся асоциальных и деструктивных взглядов, а также падкие на получение быстрой прибыли в случае финансовых затруднений могут в числе первых заняться инсайдерством.

Способ 6 – Анализ защищенности пользователей информационных систем от атак с применением социальной инженерии. Так, близкие личностные связи двух сотрудников могут позволить внешнему нарушителю воздействовать на второго (например, администратора сети) через первого (например, члена группы поддержки). Как результат, оба этих сотрудника могут непреднамеренно стать инсайдерами.

Способ 7 – Оценка потенциала сотрудника для реализации атаки, которая позволяет выявить обладающих критериями, подходящими для ведения инсайдерской деятельности. Например, высокий уровень технической подготовленности, участие в Pentest-проектах (т.е. в качестве «белого хакера») и предыдущая работа в неблагонадежных организациях (в частности, замешанных в преступной деятельности) может считаться признаками, характеризующими сотрудника как способного к успешному совершению неправомерных действий.

Модель комбинирования

Согласно вышеприведенному краткому описанию способов, каждый из них строится на собственных функциональных элементах, т.е. описывается в их терминах, точно совпадающих или обобщаемых в более абстрактное понятие. Например, Способы 5 и 7 строятся на элементе «тестирование» применительно к сотрудникам – тем самым, функциональные элементы этих способов совпадают. Аналогично, Способ 1 основывается на «событиях в жизни», а Способ 6 учитывает взаимодействие «личности» с «обществом» – все эти элементы могут быть обобщены в понятие «социума». Таким образом, логично было бы характеризовать все способы на основе одного набора элементов, т.е. в едином базисе. Тогда комбинация способов также будет работать на этом базисе, что может считаться специальной *моделью комбинирования*.

Одной из основных проблем синтеза новых базисов является их корректность в виде ортогональности – так, чтобы каждый базис не являлся бы комбинацией других. Для этого воспользуемся аппаратом категориального деления, обозначаемого vs (*аббр. от лат.* versus, *пер. на русск.* против), хорошо зарекомендовавшим себя для подобного рода методологических задач [13]. Для этого выделим 3 философские категориальные пары антагонистов, обозначаемые P (*аббр. от лат.* pair, *пер. на русск.* пара), отражающих подходы к выявлению инсайдеров. Именно эти категории и будут набором базисов, создавая тем самым 3-мерное категориальное пространство – XYZ. Каждая же точка в этом пространстве будет характеризовать принцип работы одного из 7 способов – она может быть задана набором 3-х элементов-антагонистов каждой из пар.

Авторский опыт, поверенный авторитетными публикациями других ученых-специалистов [14, 15], позволил выбрать следующие категориальные пары, характеризующие способы выявления инсайдеров и отвечающие на вопросы:

1) выявляет ли способ отклонения (***A***, *аббр. от англ.* Anormal) от нормы (***N***, *аббр. от англ.* Normal) по признакам – пара ***PX***: Нормальный (X_N) ***vs*** Аномальный (X_A);
2) анализирует ли способ постоянные (***S***, *аббр. от англ.* Static) или изменяющиеся во времени (***D***, *аббр. от англ.* Dynamic) признаки – пара ***PY***: Статический (Y_S) ***vs*** Динамический (Y_D);
3) получает ли способ признаки непосредственно от людей (***H***, *аббр. от англ.* Human) или же объектов (***O***, *аббр. от англ.* Object), с которыми они взаимодействуют – пара ***PZ***: Субъект (Z_H) ***vs*** Объект (Z_O).

Таким образом, характеристика каждого способа (*Method*) может быть записана точкой (X,Y,Z) в категориальном пространстве:

$$\begin{cases} Method \longrightarrow (X,Y,Z) \\ X \in \{X_N, X_A\} \\ Y \in \{Y_S, Y_D\} \\ Z \in \{Z_H, Z_O\} \end{cases} \quad ((1)$$

Так, например, Способ 1 основан на анализе событий в жизни сотрудника и построен из следующих функциональных элементов: «жизнь сотрудников», «события в жизни», «пребывание вне организации», «взаимодействие с миром». С этой позиции он может быть описан элементами категориальных пар: «нормальная деятельность» + «динамический анализ» + «признаки субъекта» – т.е. возникающие события в жизни сотрудника. Таким образом, Способу 1 соответствует точка: (X_N,Y_D,Z_H). Аналогичным образом, запишем в таком категориальном пространстве каждый из способов через его функциональные элементы в табличном виде (Таблица 1).

Анализ Таблицы 1 позволяет сделать вывод, что практически все способы хотя и имеют свою точку в категориальном пространстве, однако могут иметь отдельные одинаковые координаты категориальных пар и тождественные (или даже общие) функциональные элементы; что может быть использовано в дальнейшем для оценки их совместимости при комбинировании. Лишь два способа совпадают в категориальном пространстве с этой позиции (имеют единую точку) Способ 1 и Способ 5; что является закономерным, поскольку способы подобным образом анализируют изменение состояния человека при взаимодействии с внешним миром на предмет перехода в группу инсайдеров.

Экспертное комбинирование

Приведем далее все возможные комбинации пар способов и дадим их экспертную оценочную интерпретацию; очевидно, что 7 способов создадут 21 пару.

Способ 1 + Способ 2

Техническое противодействие инсайдерам затруднено тем, что, как правило, эти сотрудники компании хорошо знают, какое программное обеспечение

Таблица 1

Функциональные элементы и точки в категориальном пространстве способов выявления инсайдеров

Условное название способа	Функциональные элементы	Точка в категориальном пространстве
Способ 1. Анализ событий в реальной жизни	жизнь сотрудников, события в жизни, пребывание вне организации, взаимодействие с миром	(X_N,Y_D,Z_H)
Способ 2. Выявление аномалий в типовых сценариях работы пользователей	сценарии работы, типовое поведение, аномальное поведение, должностные обязанности	(X_A,Y_D,Z_O)
Способ 3. Предотвращение накопления критической конфиденциальной информации	сбор информации, объем информации, критичный объем, содержимое данных	(X_A,Y_S,Z_O)
Способ 4. «Ловля на живца» («Honeypot»)	«привлекательный» объект, отслеживание доступа, размещение ресурсов, пост-анализ	(X_N,Y_S,Z_O)
Способ 5. Выявление инсайдера психодиагностическими методами	психология человека, критерии нарушителя, мотивация сотрудника, тестирование сотрудника	(X_N,Y_D,Z_H)
Способ 6. Анализ защищенности пользователей от социальных атак	внешний нарушитель, социальная инженерия, общество, личность	(X_N,Y_S,Z_H)
Способ 7. Оценка потенциала пользователя для реализации атаки	потенциал сотрудника, критерии деятельности инсайдера, возможности сотрудника, тестирование сотрудника	(X_A,Y_S,Z_H)

и с какими уязвимостями используется [16, 17], а также, какие применяются политики безопасности. Именно поэтому в последнее время много внимания уделяется не только техническим средствам борьбы с инсайдерами, но и, например, анализу событий пользователей информационной системы в реальной жизни. Способ 1 схож со Способом 2, поскольку он также выделяет аномальное поведение, но не объектов, а субъектов информационной системы. Как результат, можно разработать комплекс программного обеспечения, который будет анализировать аномальные действия с объектами в организации [18] и поведение субъектов в реальной жизни, на основании чего и делать предсказания об инсайдерской деятельности.

Способ 1 + Способ 3

Экспертный анализ показал, что Способы 1 и 3 не имеют возможности работать в комплексе. Первый способ соответствует точке (X_A, Y_S, Z_O), а второй – точке (X_N, Y_D, Z_H) в категориальном пространстве. Таким образом, у них не совпадает ни один из элементов категориальной пары, что более формально подтверждает невозможность комбинирования.

Способ 1 + Способ 4

Инсайдеры – это не всегда пользователи информационной системы, которые вынашивают долгосрочный и точный план противоправной деятельности. Иногда это люди, которые попали в трудную финансовую ситуацию, разрешение которой возможно за передачу конфиденциальной информацию «третьим лицам» в обмен за вознаграждение. Подобного рода инсайдеров можно выявлять с помощью «подставных» предложений («Honeypot») от якобы «третьих лиц» в определенные трудные моменты их жизни, которые необходимо зафиксировать и учесть. Главным минусом данной комбинации однозначно является противоречие морально-этическим нормам.

Способ 1 + Способ 5

При выявлении инсайдеров психологическими методами создаются профили пользователей информационных систем с определенными показателями, которые могут указывать на склонность к инсайдерству. Одним из показателей может быть то, как сильно сложные жизненные ситуации способны подтолкнуть человека к передаче конфиденциальной информации «третьим лицам»; для определения этого возможно провести специализированное тестирование. В результате применения данной комбинации способов станет понятно, события в реальной жизни каких пользователей нужно отслеживать в первую очередь. Такой подход подобен моделям сетевых атак, но перенесенных в психологическую область. Как следствие, удастся сэкономить ресурсы, затрачиваемые на анализ поведения всех сотрудников, часть из которых в принципе не станут инсайдерами даже в тяжелых жизненных ситуациях.

Способ 1 + Способ 6

Для результативной работы Способа 6 необходимо наличие графа межличностных связей персонала с актуальной информацией, поскольку отношения между сотрудниками могут меняться на противоположные за недели или даже часы. В интересах этого необходимо постоянное обновление данного графа на основе событий, которые происходят в реальной жизни людей, работающих в организации; в этом и заключается комбинация Способов 1 и 6.

Способ 1 + Способ 7

Комбинация способов 1 и 7 может заключаться не в простом выявлении сотрудников, обладающих высоким потенциалом для совершения инсайдерских атак, а также и тех, для кого этот потенциал хотя и является условно средним, однако которым он все равно воспользуется в определенных жизненных ситуациях.

Способ 2 + Способ 3

Если пользователь информационной системы начинает собирать (и накапливать) информацию из системы хранения данных, то, следовательно, он воспроизводит аномальный сценарий работы. Таким образом, оба способа – 2 и 3 – работают на качественно едином принципе, но используя для этого разные подходы; их же объединение, очевидно, повысит общую результативность выявления инсайдеров.

Способ 2 + Способ 4

При выявлении аномалий в типовых сценариях работы пользователя можно столкнуться со сложностью отделения его нештатного поведения от обычных отклонений в работе лояльных сотрудников, связанными с определенными событиями в системе (например, ее сбой). В таком случае, для пользователей, чей профиль находится между типичным и атипичным поведением, можно устраивать дополнительную проверку в виде «ловли на живца».

Способ 2 + Способ 5

При выявлении инсайдера психологическими методами проводится ряд тестов с их последующим анализом и определением профиля пользователя информационной системы. Целесообразно до проведения такого тестирования производить анализ атипичного пове-дения или небольших отклонений в работе пользователя информационной системы.

Способ 2 + Способ 6

В Способе 6, заключающемся в анализе защищенности пользователей от социальных атак, строится граф межличностных связей персонала, в который можно занести профили нормального поведения,

как взаимодействия между некоторыми пользователями информационной системы; например, как часто один пользователь отправляет другому сообщения, содержащие конфиденциальные данные. Если поведение будет отличаться от типичного, это позволит предположить, что пользователь перешел (или может перейти) в разряд инсайдеров. Таким образом, несмотря на отсутствие у способов общих элементов категориальных пар, они все же могут быть скомбинированы, впрочем, не повысив существенно результативность обнаружения инсайдеров.

Способ 2 + Способ 7

Объединение способов аналогично комбинации Способов 1 + 7 с тем лишь отличием, что оно хотя и выявляет инсайдеров среди пользователей с потенциалом к проведению атак, но учитывает не их жизненные ситуации, а определенные события в рамках организации. Например, нарушитель может применить свои навыки лишь при возникновении определенных и крайне «удачных» условий в организации, таких, как, например, компрометация базы паролей.

Способ 3 + Способ 4

Если в системе применяется политика безопасности, основанная на предотвращении накапливания конфиденциальной информации, то потенциальному инсайдеру может оказаться сложно заполучить ее в полном объеме. Таким образом, чтобы его деятельность не была обнаружена, он будет искать иные пути достижения своей цели. В этом случае можно воспользоваться «ловлей на живца» путем предоставления ему возможности якобы получить недостающую информацию. Когда инсайдер попробует получить к ней доступ, сработает «медовая ловушка», и он будет детектирован.

Способ 3 + Способ 5

Экспертный анализ показал, что Способы 3 и 5 не имеют возможности работать в комплексе. Способ 3 соответствует точке (X_A, Y_S, Z_O), а Способ 5 – точке (X_N, Y_D, Z_H) в категориальном пространстве. Таким образом, у них не совпадает ни один из элементов категориальных пар, что более формально подтверждает невозможность комбинирования.

Способ 3 + Способ 6

Благодаря графу межличностных связей персонала становится понятно, какие пользователи и через каких работников компании могут быть подвергнуты атаке с использованием социальной инженерии. Тогда пользователям, которые особенно подвержены такому виду атак, можно ограничивать доступ к конфиденциальной информации и отслеживать накопление ими данных, чтобы при попытке передать ее «третьим лицам» не был нанесен ущерб организации.

Способ 3 + Способ 7

Аналогично комбинации Способов 3 и 6, на основании склонности поддаваться влиянию методов социальной инженерии, можно создать систему безопасности, которая также будет изолировать часть пользователей от конфиденциальной информации, но уже на основе их потенциала к реализации атак.

Способ 4 + Способ 5

Так как при выявлении инсайдера психологическими методами очень часто совершается ошибка II рода (отвергается гипотеза о том, что сотрудник – инсайдер, хотя он таковым является), то можно существенно усовершенствовать этот способ, добавив в него «ловлю на живца». В таком случае, при проведении психологического тестирования с последующим анализом результатов и определении пригодности кандидатур будут проводиться дополнительные тесты не только психологической природы, хотя и связанные с предыдущими результатами; например, системного администратора логичнее проверять на стремление к краже ключей доступа, а сотрудника бухгалтерии – на попытку разглашения финансовой информации.

Способ 4 + Способ 6

Представим, что в информационной системе существует Пользователь 1, у которого нет доступа к конфиденциальным данным, и Пользователь 2, имеющий к ним доступ. И если между такими субъектами устанавливаются тесные дружеские взаимоотношения, то становится возможным проведение атаки с использованием социальной инженерии на второго пользователя посредством первого. Иногда эту ситуацию можно решить, если предоставить Пользователю 1 мнимый доступ к информации, не предназначенной для него. В таком случае ему не придется воздействовать на Пользователя 2, а просто самому попробовать получить доступ к такой информации; в результате его инсайдерские действия будут детектированы.

Способ 4 + Способ 7

Интересным решением может стать добавление в тестирование на профпригодность при поступлении на работу механизма «ловли на живца», заключающегося в определении того, как будущий сотрудник в принципе реагирует на данные, оставленные без внимания. Так, если кандидат сходу видит бреши в системе безопасности, позволяющие ему заполучить конфиденциальную информацию, то он априори может быть опасен организации. Исключение составляют те кандидаты, которые изначально идут в отдел информационной безопасности и защиты информации и выявление подобного рода проблем входит в их компетенцию.

Способ 5 + Способ 6

Недостатком построения графа межличностных отношений персонала (Способ 6) является субъективность определения как самих связей сотрудников, так и силы их влияния. Повышение адекватности такой графовой модели (т.е. ее отражения реальной ситуации в организации) может быть осуществлено путем проведения дополнительных психодиагностических тестов (Способ 5).

Способ 5 + Способ 7

Совместное тестирование сотрудников, как с точки зрения психологии, так и с позиции потенциала для проведения атак позволит составить его более полный (и многоаспектный) «портрета, существенно повысив тем самым качество предсказания будущих инсайдеров. Так, например, признаки неуравновешенности и опыт хакерской деятельности сами по себе не будут говорить о потенциальном инсайдере, хотя их одновременное наличие у сотрудника будет крайне подозрительным, поскольку он может совершать неправомочные действия «на волне эмоций».

Способ 6 + Способ 7

Пользователи с высоким инсайдерским потенциалом вполне могут начать использовать свои социальные отношения с другими сотрудниками в целях получения через них конфиденциальной информации. Таким образом, сила связей в графе межличностных отношений может быть уточнена с учетом умения пользователей (т.е. узлов графа) проводить подобного рода атаки.

Задача комбинирования

Научно-обоснованная оценка возможности комбинирования является методологически сложной задачей. Однако для ее решения можно применить формулу (1), согласно которой каждому способу может быть поставлена в соответствие точка в категориальном пространстве. Воспользуемся для этого следующей логикой.

Во-первых, если точки способов покоординатно совпадают, это означает, что их подходы строятся на одинаковом базисе и, следовательно, способы имеют полную (*по англ.* Full) комбинируемость.

Во-вторых, если точки способов не совпадают ни по одной из координат, это означает, что их подходы строятся на абсолютно различном базисе и, следовательно, способы имеют низкую (*по англ.* Low) комбинируемость.

В-третьих, если точки способов совпадают хотя бы по одной из координат, это означает, что в их подходах совпадает один элемент категориальной пары и, следовательно, способы имеют среднюю (*по англ.* Medium) комбинируемость.

В-четвертых, если точки способов совпадают по двум координатам, это означает, что в их подходах не совпадает только один элемент категориальной пары и, следовательно, способы имеют высокую (*по англ.* High) комбинируемость.

И, в-пятых, если экспертный анализ ранее показал невозможность работы способов в комплексе, это значит, что их подходы считаются принципиально разными и, следовательно, у способов отсутствует (*по англ.* None) комбинируемость.

Исходя из введенных выше обозначений количества совпавших координат в категориальном пространстве, а также экспертного анализа комбинаций способов, дадим оценку успешности комбинирования пар; результат приведен в Таблице 2.

Табличный анализ (см. Таблицу 2) оценок возможности комбинирования способов позволяет сделать следующие выводы.

Во-первых, полная комбинируемость обнаружена для Способов 1 и 5, что закономерно, поскольку анализ жизненных событий сотрудников напрямую связан с их психоэмоциональным состоянием.

Таблица 2

Результат комбинирования пар способов выявления инсайдеров

	Способ 1	**Способ 2**	**Способ 3**	**Способ 4**	**Способ 5**	**Способ 6**	**Способ 7**
Способ 1		Medium	None	Medium	Full	High	Medium
Способ 2			High	Medium	Medium	Low	Medium
Способ 3				High	None	Medium	High
Способ 4					Medium	High	Medium
Способ 5						High	Medium
Способ 6							High
Способ 7							

Примечание. В Таблице 2 темно-серым фоном отмечены ячейки, комбинирование для которых не имеет смысла (поскольку комбинационная пара состоит из одного и того же способа). Светло-серым фоном отмечены ячейки, возможность комбинирования для которых уже указана, поскольку ячейки соответствуют таким же, но симметричным относительно диагонали. Остальные ячейки имеют следующий фон: белый для None – принципиальная невозможность комбинирования; желтый для Low – совпадение 0 координат; синий для Medium – совпадение 1-ой координаты; зеленый High – совпадение 2-х координат; красный для Full – совпадение 3-х координат.

Во-вторых, для трех следующих комбинаций способов отсутствуют совпадения их координат: Способ 1 + Способ 3, Способ 3 + Способ 5 и Способ 2 + Способ 6. При этом первые две пары не могут в принципе работать в комплексе; последняя будет иметь низкую комбинируемость, что закономерно – выявление аномалий при общении через социальные связи не принесет существенной пользы, поскольку методы социальной инженерии основаны на типовом общении людей, а не отличном от нормального.

И, в-третьих, статистика по комбинируемости имеет следующий вид: None – 2, Low – 1, Medium –10, High – 7, Full – 1. Таким образом, основная «масса» способов имеет среднюю успешность комбинирования.

Новый комбинированный «Способ 1 + 5»

Исходя из оценки возможности комбинирования способов обнаружения инсайдеров, наиболее удачной (см. Таблицу 2, значение Full) комбинацией является пара «Способ 1 + Способ 5», поскольку все их координаты в категориальном пространстве совпадают.

Суть совместной работы способов заключается в следующем. Во-первых, (следуя идее Способа 1) необходимо анализировать события в жизни сотрудников организации. Во-вторых, (следуя идее Способа 5), требуется непрерывное применение тестирования сотрудников на предмет изменений в их психологии, что может привести к совершению ими неправомерных действий (в данном случае – к инсайдерству).

Объединение идей способов позволит создать новый способ, основанный на некой модели атак на психику человека, целью которых является выведение его из лояльных сотрудников в инсайдеры. При этом источниками атак являются не инициаторы-субъекты (как в случае Способа 6), а внешние факторы. Такая модель психологических атак должна отражать особенности сотрудника, т.е. быть подстроена под него, а каждое новое событие (атака) может переводить его в следующее состояние. Достижение атакой финальной точки будет означать то, что сотрудник стал инсайдером (с психологической точки зрения).

Для обоснования работоспособности такого нового комбинированного способа приведем следующий пример. Предположим, что данная модель отражает некоторую взаимосвязь между следующими элементами: наличие дорогих гаджетов у его окружения, динамика цен на них (например, в виде графика выхода новых версий устройств), зависимость от чужого мнения и позиционирование своих интересов выше интересов компании. Очевидно, что сотрудник с высокими двумя последними показателями при повышении первых двух может попасть в ситуацию, когда резко понадобятся деньги на покупку новой версии популярного устройства. И это приведет к тому, что он с большой вероятностью попытается продать конфиденциальные данные компании третьим лицам – т.е. займется инсайдерской деятельностью.

Предложим алгоритм данной комбинации способов в формализованном виде с помощью следующего псевдокода.

```
Input:
 Environment – информация о физическом
               окружении сотрудника
 Persons[] –   информация о психологических
               особенностях всех сотрудников
 Events[] –    события, происходящие в окружении
               сотрудников
Output:
 Insiders[] –  индексы сотрудников, ставших
               инсайдерами
BEGIN
 1: VAR model = BuildModel(Environment)
 2: FOR_WITH_INDEX (person, index) IN Persons
 3:  model.SetPerson(person)
 4:  model.ApplyEvents(Events)
 5:  VAR attacks[] = model.Attacks
 6:  FOR attack IN attacks
 7:   IF attack.Rate == 100% THEN
 8:    Insiders.Add(index)
 9:    BREAK
 10:   END_IF
 11:  END_FOR
 12:  model.Reset()
 13: END_FOR_WITH_INDEX
 14: RETURN Insiders
END
```

Алгоритм на вход получает 3 параметра, определяющие психологические особенности сотрудника (Persons) и его окружения (Environment), а также динамику событий (Events) последнего.

В строке 1 по окружению (Environment) с помощью функции BuildModel строится модель (model) психологических атак (attack) на сотрудников организации.

В строке 2 начинается цикл по обходу информации о психологических особенностях каждого сотрудника (person из Persons), с указанием его индекса (index) в списке.

В строке 3 модель настраивается на текущего сотрудника (person) с помощью функции SetPerson.

В строке 4 к модели применяются произошедшие события (Events), позволяя тем самым отслеживать прогресс психологических атак (attack).

В строке 5 из модели возвращается список всех психологических атак (attack), воздействующих на сотрудника (person).

В строке 6 начинается цикл по обходу всех атак (attack из attacks) на текущего сотрудника (person).

В строке 7 проверяется диапазон (Rate) завершения текущей атаки (attack).

В строке 8, в случае завершения атак (attack) на 100% (условие в строке 7), в список инсайдеров (Insiders) с помощью функции Add() заносится

текущий индекс сотрудника (index of person), поскольку он потенциально перешел в разряд потенциальных нарушителей.

В строке 9 происходит выход из цикла по атакам (attacks), поскольку уже установлен факт инсайдерской деятельности текущего сотрудника (person).

В строке 10 завершается условие проверки завершенности атаки, начатое в строке 7.

В строке 11 завершается цикл по атакам, начатый в строке 6.

В строке 12 модель (model) с помощью функции Reset() инициализируется заново, чтобы можно было начать анализ психологического состояния следующего сотрудника (person из Persons).

В строке 13 происходит выход из цикла по сотрудникам (Persons), начатый в строке 2.

В строке 14 происходит выход из тела функции с возвратом индексов (index) всех инсайдеров (Insiders).

Таким образом, принцип работы алгоритма основан на построении общей модели психологических атак, которая считается инвариантной. Затем, для каждого сотрудника модель подстраивается под его особенности, а также производится моделирование психологических атак на основании событий, произошедших в окружении сотрудников. События могут отражать как общие изменения окружающего мира, так и события (в т.ч. и вещи), связанные с жизнью сотрудников компании. В случае если одна из атак достигла своей цели, это означает, что сотрудник потенциально перешел в разряд инсайдеров. Список таких сотрудников-инсайдеров и является результатом работы алгоритма.

Обсуждение результатов

Несмотря на очевидную, как теоретическую, так и практическую значимость проведенного исследования, сами результаты и процесс их получения обладают определенными недостатками.

Так, в работе содержится достаточно небольшое (по научным меркам) количество обзоров релевантных работ по теме комбинирования способов. Как результат, не все сделанные в секции выводы могут считаться до конца обоснованными. Однако это лишний раз подчеркивает новизну и актуальность текущей работы.

Отсутствует строгое обоснование того, что отсутствуют способы, кроме приведенных 7, или же что ни один способ не пересекается с другим. Для проверки (и в т.ч. обоснования) такой классификации можно применить уже упомянутый аппарат категориального деления. Суть аппарата заключается в том, что получаемые классы объектов обладают условием необходимости и достаточности – каждый способ будет отнесен к одному из классов, и может быть отнесен только к одному классу.

Предложенная модель комбинирования способов путем их представления в виде точки в категориальном пространстве может считаться достаточно простой (поскольку вряд ли настолько сложные способы могут быть описаны 3-мя бинарными значениями); тем не менее, это является первым шагом по формализации процесса совместимости и уже дает «строительный материал» для исследования проблем комбинирования способов [19].

Общие принципы комбинирования пар способов осуществлены, исходя из авторской точки зрения, хотя требуют более строгого (и, таким образом, научно обоснованного) подхода. Данная задача является крайне сложной и также требует дополнительного исследования. Так, например, формализация работы каждого способа и процесса их соединения в пару гипотетически позволит формализовать и итоговую комбинацию [20].

Более прагматичной, по сравнению с теоретической оценкой возможности комбинирования способов, основанной на точках в категориальном пространстве, может стать оценка и сравнение каждого из трех следующих показателей эффективности способа [21]: результативности – как меры выявления инсайдеров, оперативности – как длительности или этапа (до, во время или после атаки) их выявления, ресурсоэкономности – как объема затраченных способом ресурсов. Будем это также считать отдельно стоящей крупной задачей оценки, которая планируется к решению авторами в будущих исследованиях.

Авторы более детально описывает комбинацию Способов 1 и 5, которая по их экспертному мнению считается наиболее перспективной с позиции совместимости и величины итогового синергетического эффекта [22], что, безусловно, носит субъективный характер. Однако пример псевдокода для данной комбинации имеет определенную объективную составляющую, поскольку использует формальное описание алгоритма и основывается на строгом базисе категориальных пар.

Таким образом, несмотря на ряд недостатков (основным из которых является низкая степень формализации решений), все они имеют пути устранения.

Заключение

Первым научным результатом работы является модель комбинирования различных способов выявления инсайдеров. Новизна модели заключается в ее частичной формализации в 3-х мерном категориальном, поскольку другие способы комбинирования в основном полагаются на субъективное экспертное мнение.

Вторым научным результатом работы является экспертное и теоретическое комбинирование пар способов, позволившее получить оценки успешности такого комбинирования. Большинство комбинаций способов предложено впервые.

Третьим научным результатом работы является новый комбинированный способ, алгоритм которого задан в виде псевдокода. Аналогично, алгоритм комбинирования способов предложен впервые.

Совокупность полученных новых научных результатов позволяет сделать вывод о достижении цели исследования, а именно – сделан очередной шаг в направлении повышения эффективности противодействия инсайдерам в больших информационных системах за счет комбинаций способов их выявления.

Перспективными направлениями развития результатов настоящей работы авторы считают следующие.

1) Строгое доказательство состоятельности (необходимости и достаточности) декларированных способов выявления инсайдера за счет их классификации, полученной путем категориального деления. Не исключено, что она приведет к их агрегации с секвестрованием количества, или, наоборот – к расширению пула способов.
2) Также с научно-прогностической точки зрения интересным будет оценка возможности комбинирования всех 7 способов по аналогии с их парами. Так, одна координата каждого из способов равна примерно половине таких же координат всех остальных способов, следовательно, комбинирование всех способов имеет, по крайне мере, теоретическую вероятность.
3) Более глубокая формализация работы каждого способа, процесса их сочетания и получаемых комбинированных решений. Это позволит, с одной стороны, использовать при изучении предметной области не только эвристические, но и строго математические методы исследования, а с другой – передать комбинированное решение по выявлению инсайдеров на исполнение автомату (компьютерной программе).
4) Количественная оценка эффективности, как отдельных способов, так и комбинированных решений по критериям результативности, оперативности и ресурсоэкономности. Для этого, скорее всего, потребуется переход от мыслительных к полунатурным или имитационным экспериментам, сопровождаемым разработкой инновационных методик оценки [23].
5) Исследователи в области безопасности часто называют инсайдеров, связанных с облачными вычислениями, серьезной проблемой, но на сегодняшний день эта угроза тщательно не изучена, хотя и «озвучена» [24]. Экстраполяция полученных научных результатов в области выявления инсайдеров в больших информационных системах на цифровую облачную среду позволит по-новому ставить и решать вопросы безопасности, в том числе используя искусственный интеллект [25].

Литература

1. *Минаков С. С. Основные криптографические механизмы защиты данных, передаваемых в облачные сервисы и сети хранения 1. Ponemon Cost of Insider Threats: Global Report, 2023. URL: https://www.dtexsystems.com/resource-ponemon-insider-risks-global-report/ (дата доступа: 02.05.2024)*
2. *Verizon 2023 Data Breach Report: A Bulleted Summary. URL: https://rublon.com/blog/verizon-2023-data-breach-report-summary/ (дата доступа: 02.05.2024)*
3. *Власов Д. С. К вопросу о мотивации инсайдера организации и способах его классификации // Электронный сетевой политематический журнал «Научные труды КубГТУ». 2022. № 1. С. 128–147.*
4. *Buinevich M., Izrailov K., Kotenko I., Ushakov I., Vlasov D. Approach to combining different methods for detecting insiders // The proceedings of 4th International Conference on Future Networks and Distributed Systems (New York, USA, 2020). Iss. 26. PP. 1–6. DOI: 10.1145/3440749.3442619*
5. *Duncan A., Creese S., Goldsmith M. A Combined Attack-Tree and Kill-Chain Approach to Designing Attack-Detection Strategies for Malicious Insiders in Cloud Computing // The proceedings of International Conference on Cyber Security and Protection of Digital Services (Oxford, UK, 2019). IEEE, 2019, PP. 1–9. DOI: 10.1109/CyberSecPODS.2019.8885401*
6. *Kammüller F., Probst C. W. Combining Generated Data Models with Formal Invalidation for Insider Threat Analysis // The proceedings of Security and Privacy Workshops (San Jose, CA, USA, 2014). 2014. PP. 229–235. DOI: 10.1109/SPW.2014.45*
7. *Garfinkel S. L. Beebe N. Liu L. Maasberg M. Detecting threatening insiders with lightweight media forensics // The proceedings of International Conference on Technologies for Homeland Security (Waltham, MA, USA, 2013). IEEE, 2013. PP. 86-92. DOI: 10.1109/THS.2013.6698981*
8. *Liu A. Y., Lam D. N. Using Consensus Clustering for Multi-view Anomaly Detection // The proceedings of Symposium on Security and Privacy Workshops (San Francisco, CA, USA, 2021). IEEE, 2012. PP. 117–124. DOI: 10.1109/SPW.2012.18*
9. *Ackerman D., Mehrpouyan H. Modeling human behavior to anticipate insider attacks via System Dynamics // The proceedings of Symposium on Theory of Modeling and Simulation (Pasadena, CA, USA, 2016). 2016. PP. 1–6. DOI: 10.23919/TMS.2016.7918809*
10. *Sarma M. S., Srinivas Y., Abhiram M., Ullala L., Prasanthi M. S., Rao J. R. Insider Threat Detection with Face Recognition and KNN User Classification // The proceedings of International Conference on Cloud Computing in Emerging Markets (Bangalore, India, 2017), IEEE, 2017. PP.* 39–44, DOI: 10.1109/CCEM.2017.16.
11. *Буйневич М. В., Власов Д. С. Сравнительный обзор способов выявления инсайдеров в информационных системах // Информатизация и связь. 2019. № 2. С. 83–91. DOI: 10.34219/2078-8320-2019-10-2-83-91*
12. *Власов Д. С. Мультикритериальная модель систематизации способов обнаружения инсайдера // Вопросы кибербезопасности. 2024. № 2 (60). С. 66–73. DOI: 10.21681/2311-3456-2024-2-66-73*
13. *Буйневич М. В., Израилов К. Е., Матвеев В. В., Покусов В. В. Способ вариативной классификации уязвимостей в программном коде. Часть 1. Стратификация и категориальное деление // Автоматизация в промышленности. 2021. № 11. С. 42–49. DOI: 10.25728/avtprom.2021.11.09*
14. *Нашивочников Н. В. Выявление отклонений в поведенческих паттернах пользователей корпоративных информационных ресурсов с использованием топологических признаков // Вопросы кибербезопасности. 2023. № 4 (56). С. 12–22. DOI: 10.21681/2311-3456-2023-4-12-22.*

15. *Лебедев Д. В., Васильев Н. В. Метод выделения семантически согласованных групп пользователей социальных медиа-платформ // Техника средств связи. 2021. № 4 (156). С. 20–33.*
16. *Buinevich M., Izrailov K., Vladyko A. Metric of vulnerability at the base of the life cycle of software representations // The proceedings of 20th International Conference on Advanced Communication Technology (Chuncheon, South Korea, 2018). IEEE, 2018. PP. 1–8. URL: https://ieeexplore.ieee.org/document/8323940.*
17. *Buinevich M., Izrailov K., Vladyko A. Testing of Utilities for Finding Vulnerabilities in the Machine Code of Telecommunication Devices // The proceedings of 19th International Conference on Advanced Communication Technology (Pyeongchang, South Korea, 2017). IEEE, 2017. PP. 408–414. URL: https://ieeexplore.ieee.org/document/7890122*
18. *Поляничко М. А. Методика обнаружения аномального взаимодействия пользователей с информационными активами для выявления инсайдерской деятельности // Труды учебных заведений связи. 2020. Т. 6. № 1. С. 94–98. DOI: 10.31854/1813-324X-2020-6-1-94-98*
19. *Man D., Wang Y., Yang W., Wang W. A Combined Prediction Method for Network Security Situation // The proceedings of International Conference on Computational Intelligence and Software Engineering (Wuhan, China, 10-12 December 2010). 2010. PP. 1–4. DOI: 10.1109/CISE.2010.5676911*
20. *Lim S.-H., Yun S., Lim J., Yi O. Formalizing the design, evaluation, and analysis of quality of protection in wireless networks // Journal of Communications and Networks(). 2009. Vol. 11. No. 6. PP. 634-644. DOI: 10.1109/JCN.2009.6388417*
21. *Yu J., Oh H., Kim M., Jung S. Unusual Insider Behavior Detection Framework on Enterprise Resource Planning Systems Using Adversarial Recurrent Autoencoder // IEEE Transactions on Industrial Informatics. Vol. 18. No. 3. PP. 1541–1551. DOI: 10.1109/TII.2021.3090362*
22. *Jeridi W., Benabdallah S., Hamdi M., Boudriga N. Dynamic expert weighing for Security Risk Analysis team synergy // The proceedings of Second International Conference on Engineering System Management and Applications (Arab Emirates, 30 March 2010 - 01 April 2010). 2010. PP. 1–8.*
23. *Уткин О. В., Власов Д. С., Ильин А. В., Ефременков Е. Ю. Методика оценки деятельности должностного лица ЦУКС МЧС России // Подготовка кадров в системе предупреждения и ликвидации последствий чрезвычайных ситуаций: материалы международной научно-практической конференции. 2017. С. 227–228.*
24. *Mescheryakov S., Shchemelinin D., Izrailov K., Pokussov V. Digital cloud environment: present challenges and future forecast // Future Internet. 2020. Vol. 12. Iss. 5. PP. 82. DOI: 10.3390/fi12050082*
25. *Мадиева К. З. Искусственный интеллект и социотехнические угрозы безопасности информации // Журнал высоких гуманитарных технологий. 2024. № 1 (4). С. 38–45.*

МЕТРИКИ НА ДЕРЕВЬЯХ АТАК, СОГЛАСОВАННЫЕ С МОДУЛЬНОЙ КОМПОЗИЦИЕЙ

Волкова Е. С.[1], Гисин В. Б.[2]

DOI: 10.21681/2311-3456-2024-3-14-22

Цель исследования: *представить общую схему, в рамках которой могут быть сформулированы и вычислены метрики деревьев атак, содержащих гейты конъюнкции, дизъюнкции и секвенциальной конъюнкции.*

Методы исследования: *логико-математический анализ, линейная логика, аппарат теории категорий.*

Полученные результаты: *предложен подход к построению метрик на динамических деревьях атак, основанный на представлении метрики алгеброй над операдой деревьев атак с модульной композицией. Показано, что метрики, вычисляемые методом от листьев к корню, согласуются с модульной композицией. Наличие в дереве атаки узлов секвенциальной конъюнкции индуцирует на множестве терминальных вершин структуру направленного графа. Если этот граф ацикличен, метрики, согласованные с модульной композицией, имеют однозначную интерпретацию. При наличии на графе циклов, однозначность интерпретации обусловлена содержательными свойствами атомарных элементов атаки. В статье показано, что содержательные свойства атомарных элементов могут быть представлены соответствующими тождествами в алгебре термов. Для этого введено понятие дизъюнктивной нормальной формы динамического дерева атаки и показано, что любое дерево может быть представлено в такой форме преобразованиями, использующими только базовые тождества. Научная новизна полученных результатов состоит в применении аппарата операд для определения метрик на динамических деревьях атак.*

Ключевые слова: *дерево атак, секвенциальная конъюнкция, дизъюнктивная нормальная форма, линейная логика, модулярная категория, операда, функтор.*

COHERENT METRICS ON ATTACK TREES

Volkova E. S.[3], Gisin V. B.[4]

The purpose of research: *to present a framework within which metrics of attack trees containing conjunction, disjunction and sequential conjunction gates can be developed and calculated.*

Methods: *mathematical logic, linear logic, machinery of the category theory*

Results: *An approach to the construction of metrics on dynamic attack trees is proposed. A metric is considered as an algebra over the operad of attack trees with modular composition. Such metrics are called consistent with the modular composition. It is shown that the bottom-up calculated metrics are consistent with the modular composition. The presence of sequential conjunction nodes in the attack tree generates a directed graph on the set of the terminal vertices. If this graph is acyclic, a metric consistent with the modular composition have an unambiguous interpretation. If there are cycles on the graph, the unambiguity of interpretation is due to the substantial properties of the basic attack steps. The paper shows that the meaningful properties of atomic elements can be represented by equations in the algebra of terms. For this purpose, the concept of a disjunctive normal form of a dynamic attack tree is introduced and it is shown that any tree can be represented in this form by transformations using only basic identities. The scientific novelty of the results obtained consists in the application of operads to determine metrics on dynamic attack trees.*

Keywords: *attack tree, sequential conjunction, disjunctive normal form, linear logic, modular category, operad, functor.*

1 Волкова Елена Сергеевна, к.ф.-м.н., доцент, Финансовый университет при Правительстве Российской Федерации, Москва, Россия. E mail: evolkova@fa.ru
2 Гисин Владимир Борисович, к.ф.-м.н., профессор, Финансовый университет при Правительстве Российской Федерации, Москва, Россия. E mail: vgisin@fa.ru

3 Elena S. Volkova, Ph.D., Associate Professor, Financial University under the Government of the Russian Federation, Moscow, Russia. E mail: evolkova@fa.ru
4 Vladimir B. Gisin, Ph.D., Professor, Financial University under the Government of the Russian Federation, Moscow,. Russia. E mail: vgisin@fa.ru

Введение

Несмотря на значительный прогресс в разработке методов и средств обеспечения информационной безопасности, число инцидентов кибербезопасности существенно растет как в абсолютном, так и в относительном выражении. По данным, приведенным в материалах Positive Technologies, в IV квартале 2023 г. число кибератак выросло за год на 19%.

Вряд ли можно сомневаться в том, что дилемма «щита и меча» в сфере информационной безопасности не будет разрешена в ближайшем будущем. С учетом этого не теряет своей актуальности совершенствование и разработка новых методов оценки рисков информационной безопасности. Существует множество методов и моделей, которые были разработаны для проведения оценок безопасности.

При моделировании угроз одними из наиболее широко используемых являются модели, в основу которых положено дерево (граф) атак. Используя дерево атак, можно описать цепочки шагов атаки или уязвимостей, которые могут быть использованы злоумышленником для достижения своих целей.

Дерево атак позволяет проводить эффективный анализ безопасности путем систематической организации различных способов, с помощью которых система может быть атакована. Преимущество подобного подхода заключается в сочетании удобных для пользователя, интуитивно понятных визуальных функций с формальной семантикой и алгоритмами, позволяющими проводить качественный и количественный анализ.

Деревья атак были введены как средство представления профиля атакующего. Для достижения некоторой цели инициатор атаки может действовать в соответствии с подцелями. Существует два способа разделения цели: либо цель состоит из множества подцелей, каждая из которых должна быть достигнута; либо цель может быть достигнута с помощью одной из нескольких альтернативных подцелей. Корневой узел дерева атак представляет цель атакующего, а дочерние узлы каждого узла представляют ее уточнение до подцелей. Первоначально рассматривались дизъюнктивные (OR-узел), либо конъюнктивные (AND-узел) уточнения. Листья дерева атак представляют базовые действия атакующего и называются базовыми действиями (BAS). Дерево атак быстро стало популярным инструментом моделирования для анализа безопасности. В течение последних десятилетий графические подходы привлекли внимание многочисленных экспертов по безопасности и формальным методам и стали самостоятельной исследовательской областью (см. [1, 6, 11]).

Развиваются исследования, направленные на конструирование семантики деревьев атак. Ключевым является вопрос, когда два дерева атак могут рассматриваться как представляющие одну и ту же атаку. Если алгоритм или эксперт по безопасности модифицирует дерево атак, желательно знать семантику, чтобы понимать, инвариантны ли свойства атак относительно этих преобразований. Вообще говоря, семантика зависит от типа вопроса, для разрешения которого используется дерево атак, а вопросы характеризуются доменами атрибутов. Например, для вопросов типа «да – нет» подходит семантика, основанная на классической пропозициональной логике. Более общая семантика, основанная на мультимножествах, охватывает более широкий класс вопросов, связанных с такими атрибутами как «минимальная стоимость атаки» или «максимальный ущерб от атаки».

Однако ограничения OR-AND-формализма, в частности в отношении выражения порядка, в котором выполняются различные этапы атаки, были признаны многими авторами (см., [3, 13]). Для моделирования сценариев безопасности часто требуются конструкции, в которых должны быть четко указаны условия порядка выполнения компонентов атаки. Последовательное уточнение отличается от совместного уточнения, поскольку последнее предполагает, что злоумышленник пытается одновременно и независимо достичь нескольких подцелей. Последовательное уточнение подцелей, как и совместное уточнение, требует, чтобы были достигнуты все подцели. При этом некоторые подцели должны быть достигнуты до того, как могут быть достигнуты другие подцели. Для учета таких сценариев понятие дерева атаки было расширено, в деревьях появились узлы типа SAND (последовательные конъюнктивные уточнения).

Деревья с узлами такого типа называют динамическими. Деревья атак представляют собой в этом случае динамические описания уязвимостей системы, которые эволюционируют под влиянием развития системы и новых представлений о возможностях противника. Со временем будут добавлены новые атаки, а существующие атаки будут более конкретизированы. Эквациональная семантика позволяет только сравнить, являются ли два дерева в точности эквивалентными. Если нужно решать вопрос о том, является ли одно дерево атаки специализацией другого дерева атаки, требуется более гибкий аппарат. Для построения семантики динамических деревьев предложено использовать последовательно-параллельные графы, в которых можно представить не только действия атаки, но и причинно-следственные зависимости между действиями. Семантика тесно связана с используемым доменом атрибутов (см. [3]).

Помимо качественного анализа атак, деревья атак могут быть использованы для количественного

анализа. Обычно это делается путем присвоения оценочных значений базовым действиям и последующего расчета оценок всех остальных вершин дерева атаки вплоть до корневой. Например, каждому BAS может быть присвоено значение затрат, представляющее ресурсы, которые злоумышленник должен потратить для выполнения этого BAS, а итоговая оценка корневой вершины даст показатель минимальной стоимости успешной атаки.

Существует множество других подобных показателей, таких как минимальное время успешной атаки, среднее время компрометации, ущерб от атаки и т.п. В [8] можно найти обзор общих алгоритмов для оценки показателей безопасности, в основе которых дерево атак.

С учетом того, что имеется много различных подходов к анализу безопасности на основе дерева атак, становится актуальной разработка общей структуры, в рамках которой показатели, связанные с деревом атак, могут быть сформулированы и вычислены.

В литературе были предложены подходы к общей формализации AT-метрик (см. [4]). Они предполагают, что метрика принимает значения в полукольце. В разных работах используются разные способы определения показателя атаки в терминах базовых значений, что зачастую приводит к несовместимым определениям одного и того же показателя. Например, показатель минимального времени для динамического дерева атаки с последовательным элементом И имеет в литературе различные определения, которые несовместимы даже для небольших примеров. В частности, существует не одно определение метрик полукольца, а, по крайней мере, три несовместимых.

Несовместимость определений может быть связана с тем, что во многих работах метрика определяется вместе с алгоритмом вычисления. В результате метрики часто определяются таким образом, чтобы соответствовать алгоритму. В то же время подходы, связанные с определением метрики непосредственно на основе семантики, приводят к NP-сложным задачам [6].

Таким образом, можно констатировать, что существует потребность в формальной структуре для метрик, связанных с деревьями атак, которая была бы достаточно универсальной. В [12] предложен общий подход к определению метрик на основе операд. Необходимые и достаточные условия того, что метрика вписывается в идеологию операд, достаточно естественны. Почти все известные по публикациям метрики удовлетворяют этим условиям. В [12] детально проработано определение метрик для деревьев атак с узлами типа AND и OR и намечены пути распространения соответствующих конструкций на так называемые динамические деревья и деревья атаки-защиты. В настоящей статье мы, используя идеи из [12] и аппарат операд, даем общее определение метрик для деревьев, содержащих помимо узлов типа AND и OR также и узлы типа SAND. Заметим, что в [12] используется чрезмерно ограничительное понимание уточнения элементарного действия, которое фактически налагает запрет на использование уточняющих модулей, в которых встречаются атомарные действия, уже использованные при построении дерева атак. Чтобы обойти возникающие трудности при определении модульной композиции и сделать конструкцию достаточно общей, в работе введено понятие разметки дерева атак. Это позволяет технически разнести операцию модульной композиции и оценки дерева.

Операды являются естественным инструментом для формализации древовидных структур. Операция композиции в операдах отражает иерархическую природу деревьев. Совокупность деревьев атак можно естественным образом снабдить структурой операды. Общую идею определения когерентности метрики для деревьев атак можно представить следующим образом. Пусть D область (домен) атрибутов, в которой оцениваются BAS. Обозначим через A_n множество деревьев атаки с n терминальными вершинами, представляющими BAS. При этом два n-дерева считаются эквивалентными, если одно получено из другого путем переименования вершин и ребер Оценка терминальных вершин дерева из A_n (с фиксированной нумерацией терминальных вершин) может быть представлена элементом множества D^n. Само дерево задает отображение $D^n \longrightarrow D$, при котором оценке терминальных вершин сопоставляется оценка корневой вершины (цели атаки). Метрику можно считать согласованной (когерентной), если этим определяется морфизм операды деревьев в операду, порожденную доменом D.

Статья организована следующим образом. В следующих далее двух разделах вводятся необходимые понятия. Далее дается определение метрики, согласованной с модульной композицией, и устанавливается когерентность метрик типа «от листьев к корню». В последнем разделе описан подход к алгебраическому представлению содержательных свойств атомарных действий, позволяющий унифицировать подход, основанный на операдах.

Деревья атак

Дерево атак содержит набор терминальных узлов, структурированных с использованием операторов конъюнкции (AND) и дизъюнкции (OR). Терминальные узлы представляют атомарные действия злоумышленника (BAS). Узел AND (соответственно узел OR) считается исполненным, если исполнены все

дочерние узлы (соответственно исполнен по крайней мере один дочерний узел). Множество таких деревьев мы будем обозначать AT, а дерево атаки из AT называть AT-деревом. Множество AT-деревьев может быть расширено, если в дереве атаки допускаются узлы типа SAND. Узел SAND, считается исполненным, если исполнены все его дочерние узлы, причем в указанном порядке.

Деревья атак, содержащие узлы всех трех типов OR, AND и SAND, называют динамическими деревьями атак. Будем называть такие деревья атак DAT-деревьями.

Если не сделаны специальные оговорки, под деревом атак понимается DAT-дерево.

Более формально, дерево атак это тройка $T = (V, E, type)$, где (V, E) – дерево с корневой вершиной r_T, $type{:}V \to \{AND,OR,SAND,BAS\}$ – отображение, сопоставляющее вершинам их тип так, что выполняются следующие условия:

(1) $type(v) = BAS$ тогда и только тогда, когда v – терминальная вершина (лист);

(2) если $type(v) = SAND$, то множество дочерних узлов вершины v упорядочено.

Примечание. Здесь и далее применительно к деревьям атак мы используем термины «вершина» и «узел» как синонимы.

Для натурального числа n обозначим через $[n]$ множество $\{1,2,...,n\}$. Тогда упорядоченность из п. (2) предыдущего определения можно понимать как биекцию $\alpha{:}[n] \to ch(v)$, где n – и $ch(v)$ – соответственно число и множество дочерних узлов вершины v, а порядок определен естественным упорядочением множества $[n]$.

Через $Leaf(T)$ будем обозначать множество терминальных узлов дерева T.

Пусть задано множество Ω элементарных событий, считающихся неделимыми. Разметкой дерева атаки T будем называть отображение $\lambda{:}Leaf(T) \to \Omega$, а дерево с разметкой будем называть размеченным. Разметка «склеивает» некоторые терминальные узлы, превращая дерево T в направленный ациклический граф. Через $BAS(T)$ обозначим образ отображения разметки λ – множество элементарных событий, которые могут произойти в атаках, представленных деревом T.

Приведем ставший уже каноническим пример. На рис. 1 представлено дерево атак, направленных на кражу денег с помощью банкомата. Злоумышленник должен сначала получить соответствующие учетные данные, а затем снять деньги с банковского счета жертвы. Таким образом, корень дерева на рис. 1 уточняется с помощью метода последовательной конъюнкции (дуга со стрелкой). Чтобы получить необходимые учетные данные, злоумышленник должен украсть карту жертвы и получить соответствующий PIN-код. Порядок, в котором будут получены карта и PIN-код, не имеет значения, таким образом стандартное конъюнктивное уточнение И (простая дуга) было использовано для уточнения узла ‘получить учетные данные’. Чтобы получить PIN-код, у злоумышленника есть два варианта: он может либо провести социальную инженерию жертвы, чтобы убедить ее раскрыть секретные четыре цифры, либо найти «шпаргалку» с написанным на нем PIN-кодом. Поскольку любой из этих вариантов обеспечивает получение PIN-кода, узел «получить PIN-код» был уточнен с использованием дизъюнктивного уточнения ИЛИ (без дуги).

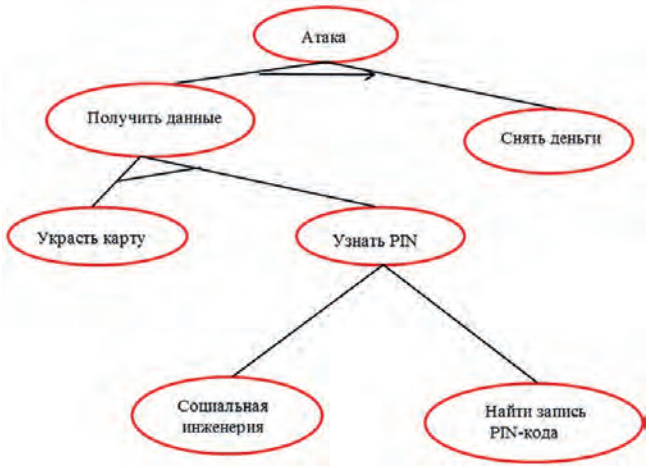


Рис. 1. Дерево атак через банкомат (1)

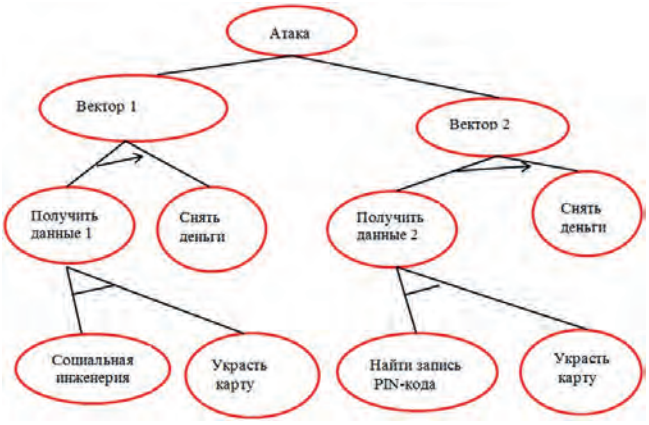


Рис. 2. Дерево атака через банкомат (2)

Дерево на рис. 2 эквивалентно дереву на рис. 1 в том смысле, что представляет те же атаки.

Размеченное дерево атак может быть представлено термом, содержащим имена базовых элементов и операторов. Например, деревья атак на рис. 1 и 2 могут быть представлены соответственно следующими термами

$$t_1 = \mathrm{SAND}(\mathrm{AND}(steal,\mathrm{OR}(eng,pin)),money), \quad (1)$$

$$t_2 = \mathrm{OR}(\mathrm{SAND}(\mathrm{AND}(eng,steal),money), \mathrm{SAND}(\mathrm{AND}(pin,steal),money)), \quad (2)$$

где eng соответствует базовому действию «Социальная инженерия», $steal$ – «Украсть карту», pin – «Найти запись PIN-кода», $money$ – «Снять деньги».

Рассмотрим отношение эквивалентности на множестве термов, порожденное системой определяющих уравнений. Пусть S_n обозначает симметрическую группу, т.е. группу автоморфизмов множества $[n]$. Тогда для любых $k,m \geq 0$ и $l \geq 1$ справедливы тождества (буквы t,u,v с индексом или без индекса служат для обозначения произвольных термов):

(B0) $\mathrm{OR}(t) = t$, $\mathrm{AND}(t) = t$, $\mathrm{SAND}(t) = t$;

(B1) $\mathrm{OR}(t,t,u_1,...,u_n) = \mathrm{OR}(t,u_1,...,u_n)$, $\mathrm{AND}(t,t,u_1,...,u_n) = \mathrm{AND}(t,u_1,...,u_n)$ для любого $n \geq 0$;

(B2) $\mathrm{OR}(t_1,...,t_n,\mathrm{OR}(u_1,...,u_m)) = \mathrm{OR}(t_1,...,t_n,u_1,...,u_m)$, $\mathrm{AND}(t_1,...,t_n,\mathrm{AND}(u_1,...,u_m)) = \mathrm{AND}(t_1,...,t_n,u_1,...,u_m)$, $\mathrm{SAND}(t_1,...,t_n,\mathrm{SAND}(u_1,...,u_m),v_1,...,v_k) = \mathrm{SAND}(t_1,...,t_n,u_1,...,u_m,v_1,...,v_k)$ для любых $n,k \geq 0$, $m \geq 1$;

(B3) $\mathrm{OR}(t_1,...,t_n) = \mathrm{OR}(t_{\sigma(1)},...,t_{\sigma(n)})$, $\mathrm{AND}(t_1,...,t_n) = \mathrm{AND}(t_{\sigma(1)},...,t_{\sigma(n)})$ для любого $n \geq 1$ и любой перестановки $\sigma \in S_n$

(B4) $\mathrm{AND}(t_1,...,t_n,\mathrm{OR}(u_1,...,u_m)) = \mathrm{OR}(\mathrm{AND}(t_1,...,t_n,u_1),...,\mathrm{AND}(t_1,...,t_n,u_m))$, $\mathrm{SAND}(t_1,...,t_n,\mathrm{OR}(u_1,...,u_m),v_1,...,v_k) = \mathrm{OR}(\mathrm{SAND}(t_1,...,t_n,u_1,v_1,...,v_k),...,\mathrm{SAND}(t_1,...,t_n,u_m,v_1,...,v_k))$ для любых $n,k \geq 0$, $m \geq 1$.

Приведенные выше тождества будем называть базовыми.

Используя базовые тождества, несложно показать, что термы (1) и (2) эквивалентны.

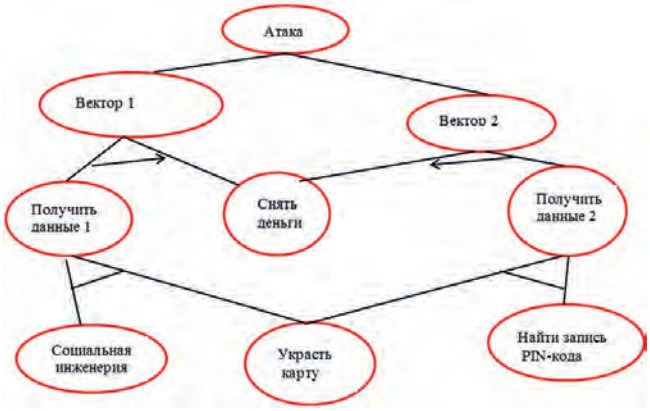


Рис. 3. Граф атака через банкомат

Перечисленные тождества порождают конфлюэнтную систему перезаписи. Соответственно дерево атаки может быть приведено к каноническому виду.

Заметим, что множество терминальных узлов дерева атаки $Leaf(T)$ является мультимножеством с базовым множеством $\mathrm{BAS}(T)$, образованным базовыми элементами атаки.

Более того, узлы типа SAND индуцируют на множестве $Leaf(T)$ отношение частичного порядка. Пусть v – вершина дерева атаки T. Обозначим через $Leaf(v)$ множество терминальных вершин дерева T, в которые из вершины v есть путь по дугам дерева T. Более формально, можно определить $Leaf(v)$ рекурсивно. Если v – терминальная вершина, то $Leaf(v) = \{v\}$; в противном случае $Leaf(v) = \cup_{u \in ch(v)} Leaf(u)$. Очевидно, $Leaf(T)$ является объединением всех множеств $Leaf(v)$. Будем считать, что терминальная вершина a предшествует терминальной вершине b и писать $a \prec b$, если в дереве T существует узел вида $\mathrm{SAND}(u_1,...,u_n)$ такой, что $a \in Leaf(u_k)$, $b \in Leaf(u_m)$ и $k < m$.

Например, для дерева на рис. 1 имеем

$$steal \prec money,\ eng \prec money,\ pin \prec money$$

Оценить успешность атаки, представленной деревом T без узлов типа SAND, можно используя бинарные логические оценки. Пусть $\mathrm{BAS} = \{a_1,...,a_n\}$. Придадим каждому базовому элементу логическое значение «активации» $b_i = I(a_i) \in \boldsymbol{B}$, где $\boldsymbol{B} = \{0,1\}$. Вектор активации $b = (b_i) \in \boldsymbol{B}^n$ приводит к успешной атаке, если оценка по стандартным правилам терма, соответствующего дереву T, дает значение 1.

Для адекватного логического описания деревьев, содержащих последовательную конъюнкцию, приходится использовать более сложную логику. В [6] показано, что адекватным инструментом в этом случае может быть линейная логика, введенная в свое время для описания логики компьютерных программ.

В качестве подходящей шкалы логических значений вместо булевой шкалы $\boldsymbol{B}$ может использоваться шкала $Q = \{0,¼,½,1\}$, содержащая четыре значения. При заданном векторе активации $b = (b_i)$ оценка дерева строится от «листьев к корню» по следующим правилам:

$I[X] = b_i$, если $X \in \mathrm{BAS}$ и $X = a_i$

$I[\mathrm{AND}(X,Y)] = 1$, если $I[X],I[Y] \neq 0$, и $I[\mathrm{AND}(X,Y)] = 0$ в противном случае

$I[\mathrm{OR}(X,Y)] = max(I[X],I[Y])$,

$I[\mathrm{SAND}(X,Y)] = 1$, если $I[X] \geq ½$ и $I[Y] \neq 0$

$I[\mathrm{SAND}(X,Y)] = ¼$, если $I[X] = ¼$ и $I[Y] \neq 0$

$I[\mathrm{SAND}(X,Y)] = 0$, если $I[X] = 0$ и $I[Y] = 0$

Атаку следует признать успешной, если итоговая оценка оказывается положительной.

Если деревья эквивалентны, оценки атак совпадают при любом векторе активации.

Атаки на динамическом дереве атак

Наличие узлов типа SAND делает целесообразным скорректировать понятие атаки (см. [2, 13]).

Пусть T – динамическое дерево атак с разметкой. Для вершины v положим $\mathrm{BAS}(v) = \lambda(Leaf(v))$. Таким образом, $\mathrm{BAS}(v)$ это множество элементарных событий, связанных с вершиной v и/или ее потомками.

Под атакой понимается множество базовых элементов $A \subseteq \mathrm{BAS}(T)$ вместе с отношением частичного порядка $\longrightarrow$. Содержательно $a \longrightarrow b$ можно интерпретировать как то, что a должно быть выполнено перед b. Таким образом, атака $(A,\longrightarrow)$ означает, что выполнены все базовые элементы из A и порядок их выполнения согласуется с $\prec$. Например, для дерева на рис.1

$$(\{pin,steal,money\},\{pin\longrightarrow money,steal\longrightarrow money\}) \quad (1)$$

является атакой. Атакой будет также и

$$(\{pin,steal,money\}, \{pin\longrightarrow money,steal\longrightarrow pin,steal\longrightarrow money\}) \quad (2)$$

В отличие от атаки (1) атака (2) не является минимальной: элемент отношения порядка $steal\longrightarrow pin$ можно удалить.

Отношение порядка $\prec$ должно быть согласовано с отношение порядка на множестве $(Leaf(T)$ в том смысле, что если $\lambda(a),\lambda(b) \in A$ и $\lambda(a)\longrightarrow\lambda(b)$, то $a \prec b$.

Успешность атаки A определим рекурсивно. Атака A успешна, если она достигает корневой вершины r_T. Атака достигает цели (вершины) v если:

(1) $type(v) = \mathrm{BAS}$ и $v \in A$
(2) $type(v) = \mathrm{OR}$ и атака A достигает некоторой вершины $u \in ch(v)$
(3) $type(v) = \mathrm{AND}$ и атака A достигает все вершины $u \in ch(v)$
(4) $type(v) = \mathrm{SAND}$, атака A достигает все вершины из упорядоченного множества $ch(v)=(u_1,\ldots,u_n)$, при этом, если $\lambda(a) \in A \cap \mathrm{BAS}(v_i)$ и $\lambda(b) \in A \cap \mathrm{BAS}(v_{i+1})$, то $\lambda(a)\longrightarrow\lambda(b)$.

В [4] введено понятие правильно сформированного (размеченного) дерева атак. Отношение порядка на множестве $Leaf(T)$ определяет структуру графа на множестве $\mathrm{BAS}(T)$. Считается, что $x,y \in \mathrm{BAS}(T)$ соединены дугой xy, если найдутся такие терминальные вершины $a,b \in Leaf(T)$, что $\lambda(a) = x$, $\lambda(b) = y$ и при этом b непосредственно следует за a относительно упорядочения $\prec$ на множестве $Leaf(T)$. Обозначим этот граф $G(T)$. Дерево T считается правильно сформированным, если граф $G(T)$ ациклический.

Дерево на рис. 1, очевидно, сформировано правильно. Терм

$$t = \mathrm{SAND}(\mathrm{OR}(x,y),\mathrm{OR}(y,z)) \quad (3)$$

дает пример неправильно сформированного дерева T: граф $G(T)$ содержит петлю в вершине y.

В [4] доказано, что пакеты атак на правильно сформированном дереве обладают важным свойством: расширение успешной атаки также успешно. Для неправильно сформированных деревьев это уже не так. Например, для дерева (3) атака

$$A = \{x,z\},\ \prec = \{x\longrightarrow z\}$$

успешна, а атака

$$A = \{x,y,z\},\ \prec = \{x\longrightarrow z\}$$

– нет (в последней для успешности нужно $\prec$ расширить до $\prec = \{x\longrightarrow y,\ x\longrightarrow z\}$).

Полученный в [4] результат позволяет описать семантику правильно сформированных деревьев атак с узлами типа SAND, сопоставляя узлу дерева v пакет минимальных атак, достигающих этот узел. Для получения семантики произвольных деревьев атак этого уже недостаточно. С узлом дерева атак приходится связывать пакет всех атак, достигающих этот узел.

Операды

Важной операцией над деревьями атак является модульная композиция. При модульной композиции базовый элемент атаки дерева T может быть заменен деревом атаки T'. Такая замена используется в том случае, когда нужно детализировать строение элемента атаки, первоначально считавшегося неделимым событием.

Чтобы рассчитать значения тех или иных метрик безопасности, зная дерево атак, можно поступить следующим образом. Фиксируется алгебраическая шкала D с операциями and, or, $sand$, удовлетворяющими соответствующим тождествам. Значения метрики приписываются базовым элементарным действиям, а метрика для всего дерева (представленного термом) рассчитывается от листьев к корню. Актуальным является также подход, при котором тем или иным способом выделяются обеспечивающие успех атаки векторы инициализации, а итоговая оценка получается как оптимальная в том или ином смысле. При таком подходе есть риск получить метрику, которая не согласуется с модульной композицией. Математическим инструментом для точного понимания согласованности служат операды.

Формально операцию модульной композиции можно представить следующим образом. Пусть $T = (V,E,type)$ – дерево атак и $a \in V$ – терминальная вершина, $type(a) = \mathrm{BAS}$. Далее, пусть $T = (V',E',type')$ – также дерево атак с корневой вершиной $r_{T'}$. Без потери общности можно считать, что множества вершин V и V' дизъюнктны. Модульная композиция – это дерево атак $T[T'/a] = (V'',E'',type'')$ такое, что $V'' = (V\setminus\{a\}) \cup V'$; $type''e(v) = type(v)$ при $v \in V$ и $type''(v) = type'(v)$ при $v \in V'$; множество E'' содержит все дуги из E и все дуги из E',

кроме тех, которые заканчиваются или начинаются в вершине a, и дополнено дугами вида (v,r_T) для всех $v \in V$, для которых $a \in ch(v)$.

Нумерацией дерева атаки T будем называть биективное отображение $\mu: Leaf(T) \to [n]$, где n – число терминальных вершин дерева T.

Пусть D – шкала оценок, которые могут быть приписаны вершинам дерева атаки. Метрика Θ, понимаемая в широком смысле, должна приписать оценку дереву атаки, если заданы оценки терминальных вершин. Рассмотрим оценку терминальных вершин $\gamma: Leaf(T) \to D$. Если терминальные вершины занумерованы отображением μ, метрика сопоставляет каждому набору $\bar{d} = (d_1,...,d_n) \in D^n$ значение $\varphi^{\Theta}_{T,\mu}(\bar{d}) \in D$, где d_i – значение оценки γ на элементе из $Leaf(T)$ с номером i, т.е. $d_i = \gamma(\mu^{-1}(i))$. Таким образом, дерево атак T задает отображение $\varphi^{\Theta}_{T,\mu}: D^n \to D$. Предположим, что $\mu': Leaf(T) \to [n]$ – еще одна нумерация. Тогда $\mu' = \sigma \circ \mu$ для некоторой перестановки σ на множестве $[n]$. Вектор оценок при нумерации μ' имеет вид $\bar{d}' = (d'_1,...,d'_n) \in D^n$, где $d'_{\sigma(i)} = d_i$.

Обозначим множество функций из D^n в D через $End_n(D)$. Перестановка σ на множестве $[n]$ определяет биективное отображение $\tau_\sigma: End_n(D) \to End_n(D)$ так, что

$$\tau_\sigma(\varphi)(d_1,...,d_n) = \varphi(d_{\sigma(1)},...,d_{\sigma(n)}). \quad (4)$$

для $\varphi: D^n \to D$.

Тогда

$$\varphi^{\Theta}_{T,\mu}(d_1,...,d_n) = \varphi^{\Theta}_{T,\mu}(d'_{\sigma(1)},...,d'_{\sigma(n)}) = \tau_\sigma(\varphi^{\Theta}_{T,\mu})(d'_1,...,d'_n). \quad (5)$$

Поскольку деревья T и T' различаются только нумерацией базовых событий, одинаковые оценки этих событий должны приводить к одинаковому значению меры Θ, то есть:

$$\varphi^{\Theta}_{T,\mu}(d_1,...,d_n) = \varphi^{\Theta}_{T',\mu'}(d'_1,...,d'_n). \quad (6)$$

Из (5) и (6) следует, что $\tau_\sigma(\varphi^{\Theta}_{T,\mu}) = \varphi^{\Theta}_{T',\mu'}$.

Замечание. Пусть $\Gamma: \Omega \to D$ – оценка элементарных действий (событий) из Ω. Оценку терминальных вершин $\gamma: Leaf(T) \to D$ дерева T будем называть согласованной с разметкой $\lambda: Leaf(T) \to \Omega$, если $\gamma = \Gamma \circ \lambda$.

Трактовка метрики как системы отображений $\varphi^{\Theta}_{T,\mu}$ в сочетании с модульной композицией деревьев естественным образом подводит к определению метрики на деревьях атак с использованием операд.

Общее понятие операды формулируется для объектов моноидальной категории. Мы приведем здесь (и будем использовать) понятие операды множеств.

Операда представляет собой набор $((R_n,\tau_n)_{n\geq 0}, id, *)$, где R_n – множество, $\tau_n: S_n \to \mathrm{Aut}(R_n)$ – гомоморфизм симметрической группы S_n (порядка n) в группу автоморфизмов множества R_n, $id \in R_1$, а $*$ – операция композиции, которая элементу $f \in R_n$ и набору элементов $g_i \in R_{m_i}$, $i=1,...,n$, ставит в соответствие элемент $f*(g_1,...,g_n) \in R_{m_1+...+m_n}$. При этом должны выполняться следующие условия:

(1) $id * f = f*(id,...,id) = f$

(2) пусть $n, m_1,...,m_n, k_1,1,...k_{n,m_n}$ – целые неотрицательные числа, и $f \in R_n$, $g_i \in R_{m_i}$ и $h_{i,j} \in R_{k_{i,j}}$, тогда

$$f*(g_1*(h_{1,1},...,h_{1,m_1}),...,g_n*(h_{n,1},...,h_{n,m_n})) = (f*(g_1,...,g_n)) * (h_1,1,...,h_{n,m_n}))$$

(3) пусть $n, m_1,...,m_n$ – целые неотрицательные числа, и $f \in R_n$, $g_i \in R_{m_i}$, а $\sigma_i \in S_{m_i}$, $i = 1,...,n$, тогда

$$f*(\tau(\sigma_1)(g_1),...,\tau(\sigma_n)(g_n)) = \tau(\sigma_1,...,\sigma_n)(f*(g_1,...,g_n))$$

где $\tau(\sigma_1,...,\sigma_n)$ перестановка порядка $m = m_1+...+m_n$, которая сохраняет на месте последовательные блоки из m_i элементов, переставляя элементы внутри блока, т.е. σ_1 действует на множестве $\{1,...,m_1\}$, перестановка σ_2 – на множестве $\{m_1+1,...,m_2\}$ и т.д.

(4) пусть $n, m_1,...,m_n$ – целые неотрицательные числа, и $f \in R_n$, $g_i \in R_{m_i}$, $i=1,...,n$, а $\sigma \in S_n$, тогда

$$\tau(\sigma)(f) * (g_1,...,g_n) = f*(g_{\sigma(1)},...,g_{\sigma(n)})$$

Пусть $\underline{R} = ((R_n,\tau_n)_{n\geq 0}, id, *)$ и $\underline{R}' = ((R'_n,\tau'_n)_{n\geq 0}, id', *)$ – операды. Морфизмом $F: \underline{R} \to \underline{R}'$ операды $\underline{R}$ в операду $\underline{R}'$ называется набор отображений $F: \underline{R}_n \to \underline{R}'_n$, $n = 0,1,...$, сохраняющих композицию и отображения τ, так что

$$F(f*(g_1,...,g_n)) = F(f) * (F(g_1),...,F(g_n)) \text{ и } F_n \circ \tau_n = \tau'_n \circ F_n$$

(для краткости мы опустили индексы в первом равенстве).

Ключевым примером для нас является операда деревьев атаки.

Для $n \geq 0$ обозначим через AT_n множество деревьев, имеющих n занумерованных терминальных вершин. Формально, элементами множества AT_n являются классы изоморфных деревьев (T,μ), где T дерево атаки с n терминальными вершинами, а $\mu: Leaf(T) \to [n]$ – нумерация терминальных вершин. Вообще, два дерева с нумерацией (T,μ) и (T',μ') считаются изоморфными, если деревья T и T' изоморфны, а нумерации совпадают после изоморфного отождествления.

Допуская некоторую вольность, мы будем говорить о деревьях атаки как элементах множества AT_n, иногда, опуская упоминание о нумерации, если это не ведет к недоразумениям.

Будем считать, что $AT_0 = \emptyset$ и $AT_1 = \{id\}$. Каждой перестановке σ на множестве $[n]$ соответствует биективное отображение $s_\sigma: AT_n \to AT_n$, которое перенумеровывает терминальные вершины: если $(T,\mu) \in AT_n$, то $s_\sigma(T,\mu) = (T,\sigma \circ \mu)$.

Роль операции * играет модульная композиция деревьев. Пусть $(T,\mu) \in AT_n$ и $(T_i,\mu_i) \in AT_{m_i}$, $i = 1,...,n$. Положим

$$T' = T * (T_1,...,T_n) = T[T_1/a_1,...,T_n/a_n] \quad (7)$$

где a_1 – терминальная вершина дерева T, для которой $\mu(a_i) = i$. Тип вершин композиции (7) сохраняется для всех нетерминальных вершин дерева T и всех вершин деревьев $T_1,...,T_n$. Иными словами, для вершины v дерева T' из (7) имеем: $type'(v) = type(v)$, если v – нетерминальная вершина дерева T, и $type'(v) = type_i(v)$, если v – вершина дерева T_i. Таким образом, множество терминальных вершин дерева T' – это объединение терминальных вершин всех деревьев $T_1,...,T_n$. Естественно, $T' \in AT_m$, где $m = m_1 +...+ m_n$.

Нумерация μ' вершина дерева T' определяется следующим образом: если a – терминальная вершина дерева T' и $a \in Leaf(T_i)$, то

$$\mu'(a) = m_1 +...+ m_{i-1} + \mu_i(a)$$

Замечание. Предполагается, что множества вершин разных деревьев дизъюнктны. Чтобы избежать ненужных сложностей, мы говорим о вершинах дерева T' как о вершинах деревьев, из которых они «родом».

Легко видеть, что модульная композиция удовлетворяет всем условиям из определения операды. Операду деревьев атаки мы будем обозначать $\underline{AT}$.

Вторым примером для нас служит операда $\underline{\mathrm{End}}(D)$. Эта операда образована семейством множеств $\mathrm{End}_n(D)$. Отображения τ определены формулой (4), единицей id служит тождественное отображение $1_D{:}D \longrightarrow D$ из $\mathrm{End}_1(D)$, а операция композиции определена композицией отображений. Заметим, что D^0 – одноточечное множество, так что можно считать, что $\mathrm{End}_0(D) = D$.

Метрику со шкалой D будем называть согласованной с модульной композицией, если она задается морфизмом операд $\underline{\mathrm{AT}} \longrightarrow \underline{\mathrm{End}}\,(D)$.

Рассмотрим метрики, когда в качестве шкалы берется множество D, на котором заданы три ассоциативные бинарные операции $\nabla,\Delta,\Diamond$ такие, что ∇ и Δ коммутативны, а Δ и $\Diamond$ дистрибутивны относительно ∇. Пусть T – дерево атак, имеющее n занумерованных терминальных вершин, а $\gamma{:}Leaf(T) \longrightarrow D$ – оценка терминальных вершин в шкале D. Определим рекурсивно $\varphi_T(\gamma) \in D$:

$$\varphi_T(\gamma)(v) = \gamma(v), \text{ если } type(v) = \mathrm{BAS}$$
$$\varphi_{T,\gamma}(v) = \nabla_{u \in ch(v)}\, \varphi_{T,\gamma}(u), \text{ если } type(v) = \mathrm{OR}$$
$$\varphi_{T,\gamma}(v) = \Delta_{u \in ch(v)}\, \varphi_{T,\gamma}(u), \text{ если } type(v) = \mathrm{AND}$$
$$\varphi_{T,\gamma}(v) = \Diamond_{u \in ch(v)}\, \varphi_{T,\gamma}(u), \text{ если } type(v) = \mathrm{SAND}$$

Наконец, $\varphi_T(\gamma) = \varphi_{T,\gamma}(r_T)$ – значение $\varphi_{T,\gamma}$ в корневой вершине дерева T. Про метрику $\gamma \mapsto \varphi_T(\gamma)$ будем говорить, что она получена методом от листьев к корню и является bu-метрикой (от bottom-up).

Соответствие $T \mapsto \varphi_T$ задает отображение $F_n{:}AT_n \longrightarrow \mathrm{End}_n(D)$. Следующая теорема утверждает, что отображения F_n определяют морфизм $\underline{\mathrm{AT}} \longrightarrow \mathrm{End}(D)$.

Теорема. Любая bu-метрика согласована.

Доказательство достаточно очевидно. Проверка того, что необходимые условия выполняются, может быть проведена прямым вычислением.

В качестве примера рассмотрим метрику, определяющую минимальное время атаки. Пусть $D = \boldsymbol{R}_{\geq 0}$ – множество неотрицательных действительных чисел,

$$x \nabla y = min(x,y),\ x \Delta y = max(x,y),\ x \Diamond y = x+y$$

(события, связанные с узлом AND могут быть выполнены параллельно, с узлом SAND – только последовательно).

Тогда, например, для дерева на рис. 2 при нумерации вершин слева направо имеем

$$\varphi_T(\gamma) = min(max(\gamma_1,\gamma_2) + \gamma_3,\ max(\gamma_4,\gamma_5) + \gamma_6)$$

При этом, очевидно, должны выполняться равенства $\gamma_2 = \gamma_5$ и $\gamma_3 = \gamma_6$ поскольку в первом случае обе оценки относятся к одному и тому же элементарному событию «украсть карту», а во втором – «снять деньги».

Тождества в алгебре термов

То, что дерево атак после разметки терминальных вершин фактически превращается в направленный граф без циклов, налагает ограничения на применимость метрики, согласованной с модульной композицией. Если дерево T правильно сформировано, т.е. граф $G(T)$ ациклический, то проблем не возникает. Если же на графе $G(T)$ имеются циклы, bu-оценка может оказаться неадекватной.

Рассмотрим, например, дерево атак T, представленное следующим термом:

$$t = \mathrm{SAND}(\mathrm{AND}(X,Y),\mathrm{AND}(X,Z))$$

В соответствии с bu-оценкой (для минимального времени) получаем:

$$\varphi_T(\gamma) = max(\gamma(X),\gamma(Y)) + max(\gamma(X),\gamma(Z)).$$

При $\gamma(X) > \gamma(Y)$ и $\gamma(X) > \gamma(Z)$ это приводит к $\varphi_T(\gamma) = 2\gamma(X)$. В то же время, если для исполнения Z не требуется повторного исполнения X, то $\varphi_T(\gamma) = \gamma(X) + \gamma(Z)$. Фактически последнее означает, что используется тождество

$$\mathrm{SAND}(\mathrm{AND}(X,Y),\mathrm{AND}(X,Z)) = \mathrm{SAND}(\mathrm{AND}(X,Y),Z). \quad (8)$$

и bu-оценка строится для его правой части.

Это тождество отражает содержательные связи между элементарными событиями. Подобными тождествами может быть представлено «знание» о связи событий.

Будем говорить, что дерево T представлено в дизъюнктивной нормальной форме (ДНФ), если выполняются следующие условия:

(1) корень r_T – единственная вершина типа OR;
(2) если v – вершина типа AND, то все вершины из $ch(v)$ имеют тип SAND;
(3) если v – вершина типа SAND, то все вершины из $ch(v)$ имеют тип AND;

Используя базовые тождества, любое дерево атак можно представить в виде ДНФ. Это утверждение доказывается стандартным образом по индукции. Выполнение условия (1) обеспечивается применением тождеств (B2) и (B4), условий (2) и (3) – тождеств (B2).

Таким образом, можно ограничиться рассмотрением метрик для деревьев, представленных в ДНФ.

Дополнительные тождества в этом случае будут связывать чередующиеся операции AND и SAND, типа тождества (8).

Например, если базовые события действия такие, что однократное исполнение события не требует его повторного исполнения в последующем, то циклы на графе $G(T)$ можно исключить, введя соответствующий набор тождеств. Тождества этого набора могут быть построены по следующей схеме.

Если в формуле имеется терм вида $\mathrm{SAND}(X_1,\ldots,X_n)$ такой, что $X \in Leaf(X_k) \cap Leaf(X_m)$ при $k < m$, то X может быть исключен из всех термов, составляющих X_m.

Например,

$$\mathrm{SAND}(X,\mathrm{AND}(Y,\mathrm{SAND}(X,Z))) =$$
$$= \mathrm{SAND}(X,\mathrm{AND}(Y,\mathrm{SAND}(Z))) = \mathrm{SAND}(\mathrm{AND}(Y,Z)).$$

Таким образом, получающееся многообразие термов служит алгебраическим эквивалентом содержательного свойства базовых действий.

Заключение

В работе предложен подход к определению метрик на динамических деревьях атак. Выделен класс метрик, которые согласуются с модульной композицией деревьев. Это метрики, которые интерпретируются как морфизмы операды деревьев в операду, порожденную измерительной шкалой как объектом соответствующей симметричной моноидальной категории. Показано, что с модульной композицией согласуются метрики, вычисляемые от листьев к корню. Наличие на дереве атак узлов с последовательной конъюнкцией типа SAND может в некоторых случаях привести к оценкам, имеющим неоднозначную интерпретацию. Добиться однозначности можно за счет более полного содержательного анализа базовых элементарных действий. Такой анализ может быть выражен алгебраически как система тождеств в алгебре термов, представляющих деревья атак. Задачей дальнейшего исследования может быть дальнейшее формирование унифицированного подхода к определению метрик на деревьях атак. В частности, описание классов деревьев атак, для которых имеются нетривиальные и практически значимые метрики, согласованные с модульной композицией.

Литература

1. *Agyepong E. Cherdantseva Y., Reinecke P., Burnap P. Challenges and performance metrics for security operations center analysts: a systematic review //Journal of Cyber Security Technology. – 2020. – Т. 4. – №. 3. – С. 125–152.*
2. *Ali A. T., Gruska D. Dynamic attack trees methodology // 2022 Interdisciplinary Research in Technology and Management (IRTM). – IEEE, 2022. – С. 1–9.*
3. *Bossuat A., Kordy B. Evil Twins: Handling Repetitions in Attack–Defense Trees: A Survival Guide //Graphical Models for Security: 4th International Workshop, GraMSec 2017, Santa Barbara, CA, USA, August 21, 2017, Revised Selected Papers 4. – Springer International Publishing, 2018. – С. 17–37.*
4. *Budde C. E., Stoelinga M. Efficient algorithms for quantitative attack tree analysis //2021 IEEE 34th Computer Security Foundations Symposium (CSF). – IEEE, 2021. – С. 1-15.*
5. *Buldas, A., Gadyatskaya, O., Lenin, A., Mauw, S., & Trujillo-Rasua, R. Attribute evaluation on attack trees with incomplete information // Computers & Security. – 2020. – Т. 88. – С. 101630.*
6. *Eades III H., Jiang J., Bryant A. On linear logic, functional programming, and attack trees //Graphical Models for Security: 5th International Workshop, GraMSec 2018, Oxford, UK, July 8, 2018, Revised Selected Papers 5. – Springer International Publishing, 2019. – С. 71–89.*
7. *Федорченко Е. В., Котенко И. В., Федорченко А. В., Новикова Е. С., Саенко И. Б. Оценивание защищенности информационных систем на основе графовой модели эксплойтов //Вопросы кибербезопасности. – 2023. – №. 3. – С. 23–36.*
8. *Konsta, A. M., Lafuente, A. L., Spiga, B., & Dragoni, N. Survey: Automatic generation of attack trees and attack graphs //Computers & Security. – 2024. – Т. 137. – С. 103602.*
9. *Lallie H. S., Debattista K., Bal J. A review of attack graph and attack tree visual syntax in cyber security //Computer Science Review. 2020. Т. 35. С. 100219. https://doi.org/10.1016/j.cosrev.2019.100219*
10. *Lopuhaä-Zwakenberg M., Budde C. E., Stoelinga M. Efficient and Generic Algorithms for Quantitative Attack Tree Analysis // IEEE Transactions on Dependable and Secure Computing. 20(5). 2022. – 4169–4187. DOI: 10.1109/TDSC.2022.3215752*
11. *Lopuhaä-Zwakenberg M., Stoelinga M. Attack time analysis in dynamic attack trees via integer linear programming //International Conference on Software Engineering and Formal Methods. – Cham : Springer Nature Switzerland, 2023. – С. 165–183.*
12. *Lopuhaä-Zwakenberg M. Attack tree metrics are operad algebras //arXiv preprint arXiv:2401.10008. – 2024.*
13. *Wu, Z., Hu, J., Zhang, X., & Ren, W. timeTree: How to Represent Time Sequence in a Threat Tree //2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys). – IEEE, 2022. – С. 2373–2378.*
14. *Zeng J., Wu, S., Chen, Y., Zeng, R., & Wu, C.Survey of attack graph analysis methods from the perspective of data and knowledge processing // Security and Communication Networks. 2019. Т. 2019. Article ID 2031063, 16 С., 2019. https://doi.org/10.1155/2019/2031063*

ПРИМЕНЕНИЕ ЛОГИКО-ВЕРОЯТНОСТНОГО МЕТОДА В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ. Часть 4

Калашников А. О.[1], Аникина Е. В.[2],Бугайский К. А.[3], Бирин Д. С.[4], Дерябин Б. О.[5], Цепенда С. О.[6], Табаков К. В.[7]

DOI: 10.21681/2311-3456-2024-3-23-32

Цель исследования: *адаптация логико-вероятностного метода оценивания сложных систем к задачам построения систем защиты информации в многоагентной системе.*

Метод исследования: *при проведении исследования использовались основные положения методологии структурного анализа, системного анализа, теории принятия решений, методов оценивания событий при условии неполной информации, логико-вероятностных методов.*

Полученный результат: *данная статья продолжает рассмотрение вопросов информационной безопасности на основе анализа отношений между субъектами и объектом защиты. Показано, что состояние отношений агента может быть получено на основе соответствующих оценок состояний на уровне информационных ресурсов и информационных потоков. Показано, что оценка состояний может быть проведена как на качественном, так и на количественном уровнях, на основе формируемых в агенте, в результате внешних воздействий, наборов событий и сообщений. Полученные результаты обеспечивают обоснованное вычисление и применение вероятностных характеристик для последующего применения логико-вероятностного метода при анализе указанных отношений.*

Научная новизна: *показана возможность определения количественных и качественных оценок состояния агента на основе формируемых в процессе функционирования событий и сообщений. Разработаны методы оценивания состояний отношений на уровне информационных ресурсов и информационных потоков через уровень доверия. Определена нижняя оценка уровня доверия к нахождению объекта в определенном состоянии. Исследованы соотношения между событиями и сообщениями из состава шаблонов состояний и текущего набора, что может быть использовано в качестве критериев при проектировании соответствующих подсистем ИС и их компонент с точки зрения информационной безопасности.*

Вклад авторов: *Калашников А. О. выполнил постановку задачи и общую разработку модели применения логико-вероятностного метода в информационной безопасности. Бугайский К. А. и Аникина Е. В. участвовали в подготовке всех разделов статьи. Бирин Д. С. и Дерябин Б. О. участвовали в подготовке раздела о формировании меры доверия. Цепенда С.О. и Табаков К.В. участвовали в подготовке раздела о доверии к состоянию объекта.*

Ключевые слова: *модель информационной безопасности, оценка сложных систем, логико-вероятностный метод, теория отношений, системный анализ*

APPLICATION OF THE LOGICAL-PROBABILISTIC METHOD IN INFORMATION SECURITY. Part 4

Kalashnikov A. O.[8], Anikina E. V.[9],Bugajskij K. A.[10], Birin D. S.[11], Deryabin B. O.[12], Tsependa S. O.[13], Tabakov K. V.[14]

1 Калашников Андрей Олегович, доктор технических наук, главный научный сотрудник лаборатории «Сложных сетей» ФГБУН Институт проблем управления им. В. А. Трапезникова РАН, г. Москва, Россия. E-mail: aokalash@ipu.ru
2 Аникина Евгения Владимировна, научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: ajanet@ipu.ru
3 Бугайский Константин Алексеевич, младший научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: kabuga@ipu.ru
4 Бирин Денис Сергеевич, младший научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: birin@phystech.edu
5 Дерябин Богдан Олегович, младший научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: baga_d@mail.ru
6 Цепенда Сергей Олегович, младший научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: tsepende.s@gmail.com
7 Табаков Кирилл Викторович, младший научный сотрудник, институт проблем управления им. В. А. Трапезникова РАН. E-mail: tabakov2002@mail.ru

8 Andrey O. Kalashnikov, Dr.Sc., Chief Scientist of the Laboratory «Complex networks» Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia. E-mail: aokalash@ipu.ru
9 Eugenia V. Anikina – research fellow, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. E-mail: ajanet@ipu.ru
10 Konstantin A. Bugajskij, Junior Researcher of the Laboratory «Complex networks» Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia. E-mail: kabuga@ipu.ru
11 Denis S. Birin – junior researcher, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. E-mail: birin@phystech.edu
12 Bogdan O. Deryabin – junior researcher, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. E-mail: baga_d@mail.ru
13 Sergey O. Tsependa – junior researcher, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. E-mail: tsepende.s@gmail.com
14 Kirill V. Tabakov 3 junior researcher, V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. E-mail: tabakov2002@mail.ru

The purpose of the article: *adaptation of the logical-probabilistic method of evaluating complex systems to the tasks of building information security systems in a multi-agent system.*

Research method: *during the research, the main provisions of the methodology of structural analysis, system analysis, decision theory, methods of evaluating events under the condition of incomplete information were used.*

The result: *this article continues the consideration of information security issues based on the analysis of the relationship between the subjects and the object of protection. It is shown that the state of the agent's relations can be obtained on the basis of appropriate assessments of states at the level of information resources and information flows. It is shown that the assessment of states can be carried out at both qualitative and quantitative levels, based on sets of events and messages formed in the agent as a result of external influences. The obtained results provide a reasonable calculation and application of probabilistic characteristics for the subsequent application of the logical-probabilistic method in the analysis of these relations.*

Scientific novelty: *consideration of information security issues using the apparatus of mathematical and logical relations. The possibility of determining quantitative and qualitative assessments of the agent's condition based on events and messages generated in the process of functioning is shown. Methods for assessing the state of relations at the level of information resources and information flows through the level of trust have been developed. A lower estimate of the level of confidence in finding an object in a certain state has been determined. The relationships between events and messages from the state templates and the current set are investigated, which can be used as criteria in the design of the corresponding IS subsystems and their components from the point of view of information security.*

Keywords: *information security model, assessment of complex systems, logical-probabilistic method, theory of relations, system analysis.*

Введение

Данная статья является четвертой из серии публикаций, посвященных исследованию вопроса применения логико-вероятностного метода при изучении вопросов защиты информации. Метод был разработан Рябининым И. А. [1, см. ссылки на соответствующую литературу там же] и приобрел популярность при проведении исследований, в том числе, связанных с анализом и оценкой рисков сложных систем. Прежде всего для решения вопросов оценки надежности работы систем и анализа причин возникновения аварийных ситуаций. Логико-вероятностный метод предполагает решение следующих задач:

1. Построение структурно-логической модели системы за счет выделения и использования событий с несовместными исходами.
2. Проведение преобразований полученных логических уравнений на основе функций булевой алгебры с целью получения системы уравнений с конечным числом переменных.
3. Теоретически обоснованный переход от уравнений булевой алгебры к уравнениям с вероятностными переменными.

К несомненным достоинствам логико-вероятностного метода следует отнести его способность обеспечить прозрачность процедур анализа и оценки сложных систем, а также хорошие адаптационные способности к новым задачам. Результатом применения логико-вероятностного метода являются количественные оценки риска как вероятности нарушения работоспособности системы. Интерес к логико-вероятностному методу – помимо типичных вопросов надежности систем, – в настоящее время подкрепляется исследованием задач машинного обучения и связанных с ними проблем оптимизации расчетов [см., например, 2–5]. В частности, логико-вероятностный метод обеспечивает хорошую точность и стабильность результатов в задачах распознавания тех или иных объектов. Логико-вероятностный метод также находит свое применение и при решении задач защиты информации [см., например, 6–11].

Тем не менее, представляется, что логико-вероятностный метод обладает значительно большим, пока не раскрытым, потенциалом в случае его дальнейшего развития и адаптации к решению различных задач в области информационной безопасности (далее – ИБ).

Постановка задачи

Логико-вероятностный метод обладает достаточно обширным набором подходов и решений по работе с логическими функциями, описывающими функционирование сложных систем, какими являются современные информационные системы (далее – ИС). В рамках достижения общей цели исследования (адаптации логико-вероятностного метода для решения задач ИБ) возникает задача разработки формально-логических основ для вычисления вероятностных параметров характеризующих истинность логических высказываний Разработка таких вероятностных параметров на системном уровне выполнена в настоящей статье.

Общие положения

Приведем основные положения предыдущих статей цикла [12–14], которые необходимы для решения поставленной задачи.

1. События и сообщения (далее – события) являются следствием внешнего воздействия на агента со стороны других агентов, участвующих в обмене информацией (далее – респондентов). Эти события образуют множество ME и определяют состояние отношений между агентом β и его респондентом γ. Такие состояния отношений $\beta R\gamma$ агента с респондентом (далее – состояния) было предложено представлять множеством $R = \{Lr,Dr,Ir,Ur\}$, где Lr означает Лояльное, Dr – Нелояльное, Ir – Неопределенное и Ur – Безразличное.
2. Агент представляет из себя набор информационных ресурсов (далее – ИР) и информационных потоков (далее – ИП), обеспечивающих обработку определенной категории данных в интересах субъекта, представленного аккаунтом ИС. В дальнейшем все ИП и ИР агента будем определять как объекты. Обозначим через K множество объектов в составе агента.
3. События формируются алгоритмически и независимо каждым из объектов агента. Обозначим через M_k полный набор событий, алгоритмически предопределенный на этапе разработки объекта $k \in K$. Тогда имеем: $ME = \bigcup_K M_k$.
4. События формируются разными источниками (например, один и тот же ИР может иметь более одного журнала регистрации событий), каждый из которых может иметь собственную семантику. Для единообразного описания событий было предложено все события ортогонализировать за счет введения пространства признаков с едиными шкалами параметров для каждого признака.
5. Для каждого объекта $k \in K$ и для каждого из возможных состояний агента $r \in R$ экспертным методом определяется эталонный набор событий (далее – шаблон) в виде матрицы свертки событий: V_k^r, то есть, наборы событий образуют иерархию $V_k^r \subseteq M_k^r \subseteq ME$.

Таким образом, определение состояния отношения агент-респондент $\beta R\gamma$ как реакцию агента на внешние воздействия возможно только на основе заранее заданного и фиксированного набора событий, формируемых независимо каждым из объектов.

Здесь необходимо отметить следующие особенности наборов событий объекта:

- полные наборы событий M_k формируются разработчиками на основе их понимания необходимости и достаточности именно такого набора событий для описания функционирования объекта и его состояний, что не всегда коррелируется с описанием внешнего воздействия с точки зрения ИБ;
- шаблоны V_k^r создаются экспертами исходя из их знаний и предпочтений, а также понимания особенностей функционирования данной ИС с точки зрения защиты информации;
- как показывает практика обеспечения ИБ, практически повсеместно в процессе сопоставления некоторых событий внешним воздействиям, эксперты используют такой параметр как «значимость» события для определения того или иного состояния, что неизбежно вызывает разночтения при оценке состояний.

Следовательно, можно говорить о предопределенной неполноте информации при определении состояния агента или *вероятностном характере* определения этих состояний. В связи с чем отметим следующие проблемы, связанные с понятием «вероятность» в ИБ.

Прежде всего обозначим, что общепринятый в естественно-научной среде частотный подход к определению вероятности практически не применим в ИБ, в силу того, что невозможно обеспечить повторяемость опыта (атаки) при постоянных параметрах контролируемой среды, которая является многозадачной и многопользовательской вычислительной системой [15, 16].

С одной стороны, события, как реакция на внешнее воздействие, формируются разными и независимыми источниками, а с другой стороны – для определения состояния могут быть необходимы цепочки событий, в том числе от одного источника [17, 18]. То есть, возникает проблема трактовки терминов «зависимость/независимость» и «совместность/несовместность» событий, что принципиально для определения вероятностных характеристик отношений $\beta R\gamma$.

Для уточнения постановки задачи в данной части статьи дадим следующую формулировку:

> *В условиях предопределенной неполноты исходных данных вероятность нахождения агента в том или ином состоянии отображает степень правдоподобности вычисляемого на основании зарегистрированных событий состояния отношения агента реальному внешнему воздействию со стороны респондента.*

Из сказанного выше следует, что источником неопределенности или вероятностного характера состояния агента являются независимо формируемые каждым из объектов наборы событий. При этом за счет ортогонализации событий как шаблоны, так и текущий набор, вызванный конкретным внешним воздействием, могут быть представлены матрицами свертки событий для каждого объекта. В дальнейшем, поскольку мы будем рассматривать вероятностные характеристики объекта, то индекс, указывающий на конкретный объект агента, будем опускать.

Формирование меры доверия

Экспертный метод формирования шаблонов V_i на основании предопределенного набора событий объекта можно представить следующей схемой: формирование предположений о возможных вариантах внешнего воздействия, а затем определение набора событий, которые могут возникнуть при том или ином внешнем воздействии. То есть формирование полного набора шаблонов для объекта равносильно формированию на основе экспертных заключений всех возможных гипотез $H = \{h_1,\cdots,h_n\}$ о внешнем воздействии. Это дает основание рассматривать шаблоны V_i как гипотезы: $V_i \equiv h_i$. Поскольку формирование шаблонов выполняется экспертным методом, то речь может идти о рациональной степени уверенности в истинности гипотезы на основе некоего доказательства. Положения предыдущего раздела позволяют определить в качестве такого доказательства уровень квалификации экспертов. Что, в свою очередь, говорит об эпистемологическом характере понятия «истинность гипотезы». Вопрос определения квалификации экспертов давно и активно исследуется, поэтому мы не будем его подробно рассматривать в данном исследовании.

В рамках текущего исследования под рациональной степенью уверенности в истинность гипотезы будем понимать меру доверия. В основе которой лежит тот факт, что одно или несколько событий могут одновременно входить в несколько шаблонов, описывающих различные внешние воздействия на основе имеющегося набора событий объекта: $\Sigma|V_i| > |M_r|$. То есть, шаблоны представляют собой «неопределенные» подмножества множества событий объекта. В данном случае не используется типичный для подобных ситуаций термин «нечеткое множество» поскольку факт вхождения события в то или иное подмножества должен трактоваться однозначно, но при этом подмножества не имеют четких границ, позволяющих утверждать отсутствие пересечения этих подмножеств. То есть, за счет наличия общих элементов границы между подмножествами не могут быть определены четко. Иными словами, каждая из сформированных экспертным методом гипотез о внешнем воздействии отражает внешнее воздействие с той или иной правдоподобностью или мерой доверия.

Опыт эксплуатации современных вычислительных средств показывает, что мощность множества событий объекта $|M_r|$ всегда больше числа событий используемых для определения состояний, что дает основания сразу выделить подмножество событий $\overline{M}$ не используемых для определения состояний. Тогда определим универсум событий объекта как $U = Mr \setminus \overline{M}$, или иначе, но тождественно, как $U = \bigcup_{i=[1,N]} V_i$, где N – общее число шаблонов, описывающих состояния объекта. Поскольку $V_i \equiv h_i$, то и универсум эквивалентен набору гипотез $U \equiv H$, что позволяет привести следующие рассуждения относительно шаблонов как гипотез.

С одной стороны, чем больше событий входит в шаблон или чем он *универсальнее*, тем больше шансов, что данная гипотеза будет в той или иной степени соответствовать внешнему воздействию.

С другой стороны, чем меньше общих с другими шаблонами событий входит в данный или чем он *уникальнее*, тем более достоверно данный шаблон описывает внешнее воздействие.

Дадим аналитическое определение свойств *универсальности* и *уникальности* шаблонов и гипотез.

Универсальность шаблона или гипотезы определим как их долю в универсуме:

$$A(h_i) = \frac{|V_i|}{|U|} \quad (1)$$

Уникальность шаблона определим как долю уникальных событий в его составе. Для этого определим события, общие для данного шаблона и остальных:

$$i,j = [1,N]\ \forall j \neq i\ F(h_i) = \bigcup_j (V_i \cap V_j) \quad (2)$$

С учетом (2) доля уникальных событий в шаблоне равна:

$$B(h_i) = 1 - \frac{|F(h_i)|}{|V_i|} \quad (3)$$

Поскольку гипотезы о состоянии объекта представляют собой «неопределенные» множества, то представляется возможным определить меру доверия гипотезы как долю уникальных событий каждого из шаблонов в универсуме. Выражения (1) и (3) можно рассматривать в качестве частотного представления вероятности как универсальности, так и уникальности шаблона. Ранее в этом разделе говорилось о формировании шаблонов экспертным методом, что дает основания положить следующую последовательность действий:

- сначала эксперты формируют каждый из шаблонов, что можно представить как определение вероятности события «универсальность шаблона»;
- затем вычленяют общие для этих шаблонов событий, что можно представить как определение вероятности события «уникальность шаблона».

При этом каждый из шагов данной последовательности действий выполнятся независимо и значение одной из величин $A(h_i)$ и $B(h_i)$ не дает информации о значении другой. Следовательно, выражения (1–3) позволяют определить меру доверия для гипотезы следующим образом:

$$\mu(h_i) = \frac{|V_i| - |F(h_i)|}{|U|} \quad (4)$$

Утверждение 1. Мера доверия гипотезы $\mu(h_i)$ является вероятностной характеристикой и может рассматриваться как априорная, то есть заданная на этапе разработки, вероятность описания внешнего воздействия конкретным шаблоном на базе универсума событий.

В качестве доказательства рассмотрим следующие свойства меры доверия.

При наличии единственной гипотезы о внешнем воздействии $|V_i| / |U| = 1$ и в отсутствие не уникальных событий в гипотезе $|F(h_i)| = 0$ получаем $\mu(h_i) = 1$. В случае отсутствия в шаблоне уникальных на универсуме событий $|F(h_i)| = |V_i|$ получаем $\mu(h_i) = 0$. Таким образом, доверие к описанию внешнего воздействия $\mu(h_i) \longrightarrow 1$ по мере того, как шаблон и соответствующая ему гипотеза обеспечивают повышение универсальности и уникальности при *заданном* на этапе проектирования универсуме событий.

Отметим, что все шаблоны являются подмножествами универсума: $U = \bigcup_{i=[1,N]} V_i$, мощность которого стоит в знаменателе выражения (4). В общем случае $\sum_{i=1}^{N} |V_i| > |U|$ за счет повторного вхождения отдельных событий в разные шаблоны, но числитель выражения (4), отражающий наличие общих событий в шаблонах позволяет утверждать, что $\mu(h_i) \leq 1$. Кроме того, выражение (2) позволяет рассматривать числитель выражения (4) как подмножество $V_i' = V_i \setminus F(h_i)$. Определим долю не уникальных событий для всех шаблонов на универсуме:

$$D(h_i) = \frac{|\bigcup_{i=1}^{N} F(h_i)|}{|U|} \quad (5)$$

С учетом «неопределенного» характера подмножеств, образующих шаблоны, выражение (5) по сути является мерой доверия для гипотезы, рассматривающей только общие события универсума. Подмножества V_i' и $\bigcup_{i=1}^{N} F(h_i)$ содержат уникальные события для каждого из шаблонов и общие для всех шаблонов события, соответственно, то есть не пересекаются и с точки зрения теории вероятностей образуют группу несовместных событий в универсуме:

$$\sum_{i=1}^{N} \mu(h_i) + D(h_i) = 1 \quad (6)$$

Доказательство завершено.

Выражения (1), (3) и (5) можно применять для оценки качества как наборов событий, задаваемых на этапе проектирования, так и работы экспертов по формированию гипотез о внешнем воздействии.

Шаблоны представляют собой сформированные экспертами гипотезы о вариантах внешнего воздействия. Но в процессе функционирования агента собственно внешнее воздействие со стороны респондента не известно, что дает возможность сформулировать следующие допущения.

Д1. Внешнее воздействие вызывает формирование в качестве ответа независимо в каждом из объектов некоторого набора событий C_k, который представляет собой подмножество полного набора событий M_k, заданного на этапе разработки объекта, то есть: $C_k \subseteq M_k$.

Д2. В общем случае подмножество C_k может содержать события входящие в полный набор событий M_k, но входящие в состав универсума, объединяющего только шаблоны состояний объекта $U = \bigcup_{i=[1,N]} V_i$. Это дает основание ввести величину $C = C_k \cap U$, которую определим как «текущий набор событий».

Д3. Текущий набор событий является единым для всех возможных состояний объекта и в общем виде можно сказать, что $\forall i \; C \cap V_i \neq \emptyset$.

Д4. Будем полагать, что формирование текущего набора событий C происходит в дискретные моменты времени, между которыми этот набор не изменяется.

Текущий набор событий C представляет из себя набор доказательств внешнего воздействия. При этом соотношение с универсумом фактически характеризует универсальность доказательной базы о внешнем воздействии:

$$A(c) = \frac{|C|}{|U|} \quad (7)$$

На основании выражений (4) и (5) определим подмножество общих для всех гипотез событий из текущего набора:

$$F(c) = C \cap (\bigcup_{i=[1,N]} F(h_i)) \quad (8)$$

Тогда доля уникальных событий текущего набора, предоставляющих доказательство для всех гипотез:

$$B(c) = 1 - \frac{|F(c)|}{|U|} \quad (9)$$

Меру доверия к доказательствам, представленным текущим набором событий определим по аналогии с (4):

$$\mu(c) = \frac{|C| - |F(c)|}{|U|} \quad (10)$$

Утверждение 2. Мера доверия к доказательствам текущего набора событий $\mu(c)$ является вероятностной характеристикой и может рассматриваться как априорная, то есть заданная на этапе разработки, вероятность доказательства внешнего воздействия на базе универсума событий.

Доказательство основано на допущениях Д1 – Д4 и представленных далее свойств меры доверия $\mu(c)$.

При равенстве мощностей текущего набора и универсума $|C| / |U| = 1$ и в отсутствие в текущем наборе общих для всех гипотез событий $|F(c)| = 0$ получаем $\mu(c) = 1$. В случае отсутствия в текущем

наборе уникальных на универсуме событий $|F(c)| = |C|$ получаем $\mu(c) = 0$. Таким образом, доверие к доказательствам внешнего воздействия $\mu(c) \longrightarrow 1$ по мере того, как текущий набор событий обеспечивает повышение универсальности и уникальности при *заданном* на этапе проектирования универсуме событий.

По аналогии с выражением (5) определим долю совпадающих не уникальных событий для шаблонов и текущего набора событий: в универсуме:

$$D(c) = \frac{|F(c)|}{|U|} \quad (11)$$

Величины $\mu(c)$ и $D(c)$ определяют доли совпадающих уникальных и общих событий для шаблонов и текущего набора, то есть для не пересекающихся подмножеств, которые с точки зрения теории вероятностей образуют группу несовместных событий в универсуме, что дает:

$$\mu(c) + D(c) = 1 \quad (12)$$

Доказательство завершено.

В третьей части статьи было доказано следующее утверждение (см. [14], Утверждение 1): «*Состояние объекта определяется соотнесением текущего набора событий и эталонного набора*». Как было показано в предыдущей статье цикла [14], все события текущего набора и универсума ортогонализированы, то есть приведены к единообразному с точки зрения ИБ виду и могут быт представлены в виде матриц свертки событий.

В терминологии настоящей статьи данное утверждение можно переформулировать следующим образом.

Утверждение 3. Состояние объекта определяется выполнением операций сравнения с целью подсчета числа совпадающих элементов матрицы свертки событий текущего набора C и универсума U, представленного матрицами свертки событий шаблонов V_i.

При этом относительно универсума как набора гипотез $H = \{h_1,\cdots,h_n\}$: (в силу $V_i \equiv h_i$) необходимо отметить следующее:

- данный набор гипотез отражает все возможные внешние воздействия, то есть набор является полным;
- в каждый конкретный момент времени респондент может реализовывать только одно воздействие, соответствующее одной или нескольким гипотезам, то есть гипотезы совместны.

Из допущений *Д1 – Д4* и Утверждения 3 следует, что сравнение матриц свертки можно рассматривать как ответ объекта на внешнее воздействие, который так или иначе совпадает с каждым из шаблонов. Тогда следует предположить, что наилучшим ответом g_i на внешнее воздействие будет гипотеза h_i в наибольшей степени совпадающая с текущим набором событий. Определим события, общие для текущего набора событий и гипотезы h_i:

$$F(g_i) = C \cap V_i \quad (13)$$

Определим функцию, выражающую ответ объекта как «расстояние» между гипотезой и ответом на внешнее воздействие: $\lambda : H \times G \longrightarrow E$. В дальнейшем будем обозначать ее как $\lambda(h_i,g_i)$:

$$\lambda(h_i,g_i) = \frac{|F(g_i)|}{|V_i|} \quad (14)$$

Необходимо отметить, что из выражения (13) следует, что $|F(g_i)| \le |V_i|$, то есть выражение (14) не может иметь числитель меньший нуля.

Утверждение 4. Функция расстояния $\lambda(h_i,g_i)$ является вероятностной характеристикой и может рассматриваться как мера соответствия внешнего воздействия отдельной гипотезе о таковом воздействии, представленной эквивалентным шаблоном.

В качестве доказательства рассмотрим следующие свойства функции расстояния.

При полном совпадении матриц свертки событий текущего набора и шаблона, что в общем случае при $|C| \ge |V_i|$ тождественно $F(g_i) = V_i$, функция $\lambda(h_i,g_i) = 0$. В противном случае, когда $F(g_i) \longrightarrow \varnothing$, функция $\lambda(h_i,g_i) \longrightarrow 1$, что можно трактовать как увеличение расстояния или не совпадения элементов матриц свертки событий текущего набора и шаблона. Тогда дополнение функции расстояния до единицы можно определить как правдоподобие соответствия ответа объекта внешнему воздействию, представленному той или иной гипотезой:

$$\delta(h_i,g_i) = 1 - \lambda(h_i,g_i) \quad (15)$$

Важно отметить, что $0 < \delta(h_i,g_i) \le 1$ всегда, в силу выражения (13), которое предполагает обязательное наличие событий текущего набора и шаблона в процессе функционирования объекта.

По аналогии с выражением (5) определим долю совпадающих не уникальных событий для отдельного шаблона и текущего набора событий:

$$D(g_i) = \frac{|C \cap F(g_i)|}{|V_i|} \quad (16)$$

Несложно показать, что значение $D(g_i) \le |F(g_i)|$ и если положить $1 = |V_i| / |V_i|$ в (15), то можно определить правдоподобие ответа на гипотезу следующим образом:

$$\delta(h_i,g_i) = \frac{|F(g_i)|}{|V_i|} - D(g_i) \quad (17)$$

Доверие состояния объекта

Выражение (17) подразумевает в качестве гипотезы о внешнем воздействии некоторый шаблон, а в качестве ответа на внешнее воздействие – уникальные события текущего набора.

Напомним, что в общем виде состояние отношения является решением агента о характере и степени опасности воздействия со стороны респондента в процессе обмена данными. То есть можно говорить о решении агентом когнитивной задачи по определению сходства состояния отношения с истинными действиями респондента, которое заключается в определении вероятности нахождения агента в том или ином состоянии на основании такого сходства.

При таком подходе уникальные события текущего набора должны рассматриваться в качестве доказательств в пользу той или иной гипотезы внешнего воздействия, а каждое из возможных состояний множества R – в качестве заключения о наиболее правдоподобной гипотезе внешнего воздействия при наличии известных доказательств.

Но поскольку реальное внешнее воздействие нам не известно, то это позволяет, как следует говорить[15] только об ожидаемой достоверности определения состояния $P(r,g_i)$ как ответа на внешнее воздействие на основании Утверждения 3, и оценки правдоподобия $\delta(h_i,g_i)$. Таким образом, эти величины можно рассматривать как апостериорную эпистемологическую вероятность и оценку истинности гипотезы соответственно, что дает основание сделать следующий промежуточный вывод

- в основу определения достоверности состояния может быть положена байесовская теория принятия решений, когда достоверность определения состояния пропорциональна расстоянию (14) между гипотезой и ответом на внешнее воздействие;
- для каждого внешнего воздействия, представленного текущим набором событий C необходимо определять возможный ответ g для всех гипотез h_i в виде шаблонов V_i.

Величина $P(r,g_i)$, определяемая для каждого состояния по максимально возможному числу доказательств, может рассматриваться как потенциально наиболее правильный ответ на гипотезу о возможном реальном внешнем воздействии на объект. При условии, что данное доказательство представляется наиболее верным для данной гипотезы (Gabbay Dov M., Hartmann S., Wood J. The Development of the Hintikka Program // Handbook of the History of Logic. – 2011. – Vol. 10. – P. 311–356.). Термин «наиболее верным» на основании Утверждений 1, 2 и 4 будем определять как использование в качестве доказательств уникальных событий из состава текущего набора и шаблонов.

Как следует из выражений (6), (12) и (17), для определения величины $P(r,g_i)$ на основе байесовской теории принятия решений целесообразно дать следующие трактовки определенных ранее величин доверия и правдоподобия:

- мера доверия гипотезы $\mu(h_i)$ в качестве априорной вероятности принадлежности заданных на этапе разработки событий к определенной гипотезе о внешнем воздействии;
- мера доверия к текущему набору событий $\mu(c)$ в качестве априорной вероятности принадлежности текущего набора событий к определенному ответу на внешнее воздействие;
- истинное правдоподобие $\delta(h_i,g_i)$, как вероятность принадлежности событий текущего набора к одной из гипотез.

В итоге получаем следующее определение апостериорной вероятности правдоподобности гипотезы при наличии данных доказательств:

$$P(r,g_i) = \frac{\mu(c)\delta(h_i,g_i)}{\mu(h_i)} \qquad (18)$$

Утверждение 5. Апостериорная вероятность $P(r,g_i)$ дает оценку правдоподобия нахождения объекта в том или ином состоянии или уровень доверия к нахождению объекта в определенном состоянии.

Доказательство утверждения основано на Утверждениях 1–4, обеспечивающих вычисление степени правдоподобия ответа g объекта, представленного текущим набором событий C, гипотезам о внешнем воздействии h_i, представленными в виде шаблонов V_i, которые в свою очередь эквивалентны состояниям объекта. При этом, согласно Gabbay Dov M., Hartmann S., Wood J., апостериорная вероятность, определяемая по максимально возможному числу доказательств, может рассматриваться как потенциально наиболее правильный ответ на гипотезу о реальном внешнем воздействии на объект, если данные доказательства представляются наиболее правдоподобными для данной гипотезы.

Утверждение 6. Величина $P(r,g_i)$ является нижней оценкой уровня доверия к нахождению объекта в определенном состоянии.

Доказательство основано на том факте, что для расчета уровня доверия используются уникальные события из состава текущего набора и шаблонов, которые согласно (2–16) заведомо меньше полных составов как текущего набора событий, так и шаблонов. А поскольку формирование шаблонов, отображающих те или иные гипотезы о внешнем воздействии выполняются экспертным методом,

15 Gabbay Dov M., Hartmann S., Wood J. The Development of the Hintikka Program // Handbook of the History of Logic. – 2011. – Vol. 10. – P. 311–356.

то уникальные события образуют фиксированные на этапе разработки подмножества универсума.

В самом общем случае можно полагать, что каждый шаблон соответствует одному из состояний объекта, то есть $V_i \equiv h_i \equiv r$. Это дает возможность рассматривать величину $P(r,g_i)$ как уровень доверия для отдельных состояний $r \in R$, $R = \{Lr,Dr,Ir,Ur\}$.

Однако, внешнее воздействие в соответствии с базами данных mitre.org может относиться к разным классам по механизмам или доменам атак (CAPEC) с использованием различных типов слабых мест программного и аппаратного обеспечения (CWE) объектов из состава агента. Что дает основание представлять внешнее воздействие несколькими различными с этой точки зрения наборами шаблонов $X_i^r \subseteq M_k \subseteq ME$, $X_i = \{V_1,\cdots,V_L\}$ $i = [1,L]$ и L – число шаблонов, описывающих конкретное состояния для отдельного объекта. Несложно показать, что в рамках логико-вероятностного метода следует сделать следующее допущение.

Д5. Все шаблоны из набора, описывающего состояние объекта, образуют полную группу событий.

Тогда наборы шаблонов X^r могут рассматриваться как предикаты z_i, $i \in L$. То есть, состояние может быть представлено в виде логической формулы $r = \vee(i \in N)z_i$, что дает следующий полином $p^*(X^r) = p_1 + p_2(1 - p_1) + p_3(1 - p_2)(1 - p_1) + \cdots$, где p_i определяется согласно (18) для каждого шаблона.

Рассмотрим выражение (18), где представляет интерес отношение априорных вероятностей. С учетом выражений (4) и (10) можем записать следующее уравнение как условие получения наибольшего значения величины $P(r,g_i)$:

$$\frac{|C| - |F(c)|}{|V_i| - |F(h_i)|} = 1 \qquad (19)$$

С учетом выражений (2) и (8) на основании выражения (19) можно сделать следующие выводы:

W1. Мощность универсума не имеет принципиального значения для определения доверия к состоянию объекта.

W2. Ситуации, когда шаблон или текущий набор событий полностью состоят из общих с другими шаблонами событий, не имеют смысла при определении состояния объекта, то есть с точки зрения ИБ.

W3. Как уже отмечалось выше, $0 < \delta(g_i,h_i) \le 1$, а значит и $P(g,h) > 0$ всегда в силу (см. (13)) обязательного наличия событий текущего набора и шаблона в процессе функционирования объекта, то есть $P(r,g_i) > 0$ при $C > \Sigma_{i=1}^N F(h_i)$.

Неопределенность состояния

Как было показано в предыдущих разделах настоящей статьи, определение доверия к состоянию объекта базируется на шаблонах, представляющих ту или иную гипотезу о внешнем воздействии. Предварительно сделаем следующее допущение.

Д6. Без потери общности будем полагать, что каждое состояние объекта описывается одной гипотезой, которой соответствует один шаблон.

Все шаблоны представляют собой «неопределенные» множества, поэтому выражения (6), (12), (17) содержат величины $D(*)$, характеризующие долю не уникальных событий в шаблонах и их пересечении с событиями текущего набора. Эти общие события (2), которые не могут быть однозначно отнесены к тому или иному шаблону, создают неопределенность конкретного состояния объекта. Для определения этой неопределенности воспользуемся энтропийным подходом по аналогии с [19].

Рассмотрим условия получения предельных значений неопределенности состояния объекта на основании выражений (2), (5), (11) и (16).

Максимальная неопределенность состояния объекта $D(*) = 1$ возникает при следующих условиях:

$$\cup_{i=1}^N F(h_i) = U;$$
$$F(h) = U;$$
$$C \cap F(h_i) = V_i;$$
если $D(c) = 1$, то и $D(h_i) = 1$.

Минимальная неопределенность состояния объекта $D(*) = 0$ возникает при следующих условиях:

$$\cup_{i=1}^N F(h_i) = \varnothing;$$
$$C \cap (\cup_{i=1}^N F(h_i)) = \varnothing;$$
$$C \cap F(h_i) = \varnothing.$$

Обобщая перечисленные условия, можно вывести следующие свойства неопределенности для отдельного состояния $D(r)$.

$$D(r) = 0 : (V_i = U) \vee (F(h_i) = \varnothing \wedge (V_i \subset U)) \qquad (20)$$

$$D(r) = 1 : (F(h_i) \cap V_i = V_i) \wedge (V_i \subset U)) \qquad (21)$$

Выражение (20) означает, что минимум неопределенности состояния достигается, когда все события универсума принадлежат единственному шаблону или, когда шаблон не содержит общих со всеми другими шаблонами событий из состава универсума.

Соответственно, выражение (21) показывает, что максимум неопределенности состояния достигается, когда шаблон состоит только из общих с другими шаблонами событий.

Напомним, что $D(h_i)$ представляет собой долю общих событий для всех состояний объекта или значение общей неопределенности всех состояний, а $D(c)$ – это значение общей неопределенности для всех состояний в ответе объекта на внешнее воздействие. Соответственно, эти значения можно рассматривать как априорные и апостериорные величины. При таком подходе представляет интерес случай, когда $D(c) = D(h_i)$. На основании выражений (5) и (11) можно вывести, что $F(c) = \cup_{i=1}^N F(h_i)$

и с учетом (8) получаем $C \cap (\cup_{i=1}^{N} F(h_i) = \cup_{i=1}^{N} F(h_i)$. Из чего следует, что $C = (\cup_{i=1}^{N} F(h_i)$. На основании чего с учетом (2) и (19) можно сделать следующий вывод:

W4. Для определения состояния объекта между текущим набором событий и общими событиями отдельного шаблона V_i, описывающих гипотезы о внешнем воздействии необходимо выполнение условия:

$$\min|C| = \Sigma_{i=1}^{N} |F(h_i)| \quad (22)$$

Выражение (22) также, как и выражения (1), (3) и (5), можно применять для оценки качества как наборов событий, задаваемых на этапе проектирования, так и работы экспертов по формированию гипотез о внешнем воздействии. Соответственно, представляет интерес дальнейшее исследование соотношения шаблонов и текущего набора на основе данных выражений как риск-ориентированного критерия в ИБ.

Согласно (13) по результатам внешнего воздействия мы имеем $F(g_i) = C \cap V_i$ и соответственно, $F(g_i) \subseteq V_i$ Следовательно, целесообразно положить, что уменьшение доли событий шаблона, совпадающих с ответом объекта на внешнее воздействие, повышает неопределенность состояния. Из чего следует, что с учетом выражений (4) и (10) получаем величину, которая показывает долю неиспользованных событий из шаблона:

$$D(c,v_i) = \frac{|V_i| - |F(g_i)|}{V_i} \; (|V_i| - |F(g_i)|)/|\text{V\_i}| \quad (23)$$

На основании выражения (16) несложно показать, что по результатам внешнего воздействия $C \cap F(h_i) \subseteq F(h_i)$. Следовательно, целесообразно положить, что увеличение доли общих событий, совпадающих с ответом объекта на внешнее воздействие, повышает неопределенность состояния. Откуда по аналогии с (23) можем получить долю неиспользованных общих событий:

$$D(c,h_i) = \frac{|F(h_i)| - |C \cap F(h_i)|}{V_i} \quad (24)$$

Из (3) можем определить априори заданную на этапе разработки собственную неопределенность шаблона, которая соответствует свойству (21):

$$D(v_i) = \frac{|F(h_i)|}{V_i} \quad (25)$$

В [19] была показана возможность применения энтропийного подхода для учета неполноты информации, описывающей как гипотезы о внешнем воздействии, так и ответов на него при определении состояния объекта. Тогда выражения (23)–(25) дают основание привести следующее выражение для неопределенности состояния объекта:

$$D_r = \ln(1 + D(v_i)) - \ln(1 + D(c,h_i)) + \ln(1 + D(c,v_i)) \quad (26)$$

Рассмотрим предельные случаи, определяемые (20) и (21), для выражения (26).

Пусть $D(c,h_i) = 0$, то есть текущий набор событий включает все общие события шаблона $F(h_i) = C \cap F(h_i)$, что дает $F(h_i) \geq C \cap F(h_i)$. Для случая равенства $F(g_i) = F(h_i)$ выражение (26) может быть представлено как $[(F(h_i) + (V_i - F(h_i))] \; / \; V_i$, что дает $D_r = 1$. В случае $F(g_i) > F(h_i)$ получаем:

$$D_r = D(v_i) + D(c,v_i) \quad (27)$$

Пусть $D(c,h_i) = 1$, то есть текущий набор событий исключает все общие события шаблона $C \cap F(h_i) = \varnothing$, что дает $F(h_i) = V_i$, тогда получаем:

$$D_r = D(c,v_i).$$

Пусть $D(c,v_i) = 1$, то есть $F(g_i) = \varnothing$, что влечет $C \cap F(h_i) = \varnothing$ и $D(c,v_i) = D(v_i)$, тогда получаем:

$$D_r = 1.$$

Пусть $D(c,v_i) = 0$, то есть $F(g_i) = V_i$ и получаем, что $D(c,v_i) = 0$. Из чего следует, что для случая $F(h_i) = V_i$ имеем $D_r = 1$, а в остальных случаях $D_r = D(v_i)$.

Отметим еще раз, что $D_r = 0$ только при условии $F(h_i) = \varnothing$ и $V_i = C \cap V_i$.

Ранее отмечалось, что величину $D(v_i)$ следует рассматривать как априорно заданную на этапе разработки неопределенность состояния объекта. Если трактовать долю уникальных событий $V_i^1 = V_i \setminus F(h_i)$ как уровень подтверждения гипотезы с вероятностью равной 1, то наличие общих сообщений в шаблонах можно рассматривать как резерв, позволяющий повысить значение $P(r,g_i)$ за счет отнесения общих событий к конкретному шаблону, соответствующему конкретной гипотезе о внешнем воздействии. Подмножества V_i' и $\cup_{i=1}^{N} F(h_i)$ содержат уникальные события для каждого из шаблонов и общие для всех шаблонов события, соответственно, то есть не пересекаются и с точки зрения теории вероятностей образуют группу несовместных событий в универсуме. Тогда выражение (27) совместно с Утверждением 6 позволяет сделать следующий вывод:

W5. Уровень доверия к нахождению объекта в определенном состоянии фактически может быть определен только на интервале значений от $P(r,g_i)$ до $P(r,g_i) + P(D_r)$, где $P(D_r)$ можно рассматривать как функцию разрешения неопределенности.

Заключение

В рамках заявленной цели настоящего исследования (адаптации логико-вероятностного метода для решения задач ИБ) в статье разработаны формально-логические основы получения вероятностных оценок как результата обработки событий и сообщений, формируемых в процессе функционирования агента. Данные вероятностные оценки необходимы

для последующего определения состояний отношений агентов на основе логико-вероятностного метода при рассмотрении вопросов защиты информации в многоагентных системах. Полученные результаты показывают недостаточность собственных возможностей агента по определению состояния отношений с респондентами. Предлагаемые механизмы количественного и качественного оценивания результатов обработки событий и сообщений могут быть использованы при проектировании соответствующих подсистем современных ИС и их отдельных компонентов.

Литература

1. *Рябинин И. А. Решение одной задачи оценки надежности структурно-сложной системы разными логико-вероятностными методами / И. А. Рябинин, А. В. Струков // Моделирование и анализ безопасности и риска в сложных системах, Санкт-Петербург, 19–21 июня 2019 года. – Санкт-Петербург: Санкт-Петербургский государственный университет аэрокосмического приборостроения, 2019. – С. 159–172.*
2. *Демин А. В. Глубокое обучение адаптивных систем управления на основе логико-вероятностного подхода / А. В. Демин // Известия Иркутского государственного университета. Серия: Математика. – 2021. – Т. 38. – С. 65–83.*
3. *Викторова В. С. Вычисление показателей надежности в немонотонных логико-вероятностных моделях многоуровневых систем / В. С. Викторова, А. С. Степанянц // Автоматика и телемеханика. – 2021. – № 5. – С. 106–123.*
4. *Леонтьев А. С. Математические модели оценки показателей надежности для исследования вероятностно-временных характеристик многомашинных комплексов с учетом отказов / А. С. Леонтьев, М. С. Тимошкин // Международный научно-исследовательский журнал. – 2023. – № 1(127). С. 1–13.*
5. *Пучкова Ф. Ю. Логико-вероятностный метод и его практическое использование / Ф. Ю. Пучкова // Информационные технологии в процессе подготовки современного специалиста: Межвузовский сборник научных трудов / Министерство просвещения Российской Федерации; Федеральное государственное бюджетное образовательное учреждение высшего образования «Липецкий государственный педагогический университет имени П. П. Семенова-Тян-Шанского». Том Выпуск 25. – Липецк: Липецкий государственный педагогический университет имени П. П. Семенова-Тян-Шанского, 2021. – С. 187–193.*
6. *Россихина Л. В. О применении логико-вероятностного метода И. А. Рябинина для анализа рисков информационной безопасности / Л. В. Россихина, О. О. Губенко, М. А. Черноситова // Актуальные проблемы деятельности подразделений УИС: Сборник материалов Всероссийской научно-практической конференции, Воронеж, 20 октября 2022 года. – Воронеж: Издательско-полиграфический центр «Научная книга», 2022. – С. 108-109.*
7. *Карпов А. В. Модель канала утечки информации на объекте информатизации / А. В. Карпов // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2018): VII Международная научно-техническая и научно-методическая конференция. Сборник научных статей. В 4-х томах, Санкт-Петербург, 28 февраля – 01 марта 2018 года / Под редакцией С. В. Бачевского. Том 2. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, 2018. – С. 378–382.*
8. *Методика кибернетической устойчивости в условиях воздействия таргетированных кибернетических атак / Д. А. Иванов, М. А. Коцыняк, О. С. Лаута, И. Р. Муртазин // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2018): VII Международная научно-техническая и научно-методическая конференция. Сборник научных статей. В 4-х томах, Санкт-Петербург, 28 февраля – 01 марта 2018 года / Под редакцией С.В. Бачевского. Том 2. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2018. – С. 343–346.*
9. *Елисеев Н. И. Оценка уровня защищенности автоматизированных информационных систем юридически значимого электронного документооборота на основе логико-вероятностного метода / Н. И. Елисеев, Д. И. Тали, А. А. Облаченко // Вопросы кибербезопасности. – 2019. – № 6(34). – С. 7–16.*
10. *Коцыняк М. А. Математическая модель таргетированной компьютерной атаки / М. А. Коцыняк, О. С. Лаута, Д. А. Иванов // Наукоемкие технологии в космических исследованиях Земли. – 2019. – Т. 11, № 2. – С. 73–81.*
11. *Белякова, Т. В. Функциональная модель процесса воздействия целевой компьютерной атаки / Т. В. Белякова, Н. В. Сидоров, М. А. Гудков // Радиолокация, навигация, связь: Сборник трудов XXV Международной научно-технической конференции, посвященной 160-летию со дня рождения А. С. Попова. В 6 томах, Воронеж, 16–18 апреля 2019 года. Том 2. – Воронеж: Воронежский государственный университет, 2019. – С. 108–111.*
12. *Калашников А. О. Применение логико-вероятностного метода в информационной безопасности (Часть 1) / А. О. Калашников, К. А. Бугайский, Д. С. Бирин, Б. О. Дерябин, С. О. Цепенда, К. В. Табаков // Вопросы кибербезопасности. – 2023. – № 4 (56). – С. 23–32. DOI: 10.21681/2311-3456-2023-4-23-32*
13. *Калашников А. О. Применение логико-вероятностного метода в информационной безопасности (Часть 2) / А. О. Калашников, К. А. Бугайский, Е. И. Аникина, И. С. Перескоков, Ан. О. Петров, Ал. О. Петров, Е. С. Храмченкова, А. А. Молотов // Вопросы кибербезопасности. – 2023. – № 5 (57). – С. 113–127. С. 23–32. DOI: 10.21681/2311-3456-2023-5-113-127*
14. *Калашников А. О. Применение логико-вероятностного метода в информационной безопасности (Часть 3) / А. О. Калашников, К. А. Бугайский, Е. И. Аникина, И. С. Перескоков, Ан. О. Петров, Ал. О. Петров, Е. С. Храмченкова, А. А. Молотов // Вопросы кибербезопасности. – 2023. – № 6 (58). – С. 20–34. С. 23–32. DOI: 10.21681/2311-3456-2023-6-20-34*
15. *Калашников А. О. Инфраструктура как код: формируется новая реальность информационной безопасности / А. О. Калашников, К. А. Бугайский // Информация и безопасность. – 2019. – Т. 22, № 4. – С. 495–506.*
16. *Бугайский К. А. Расширенная модель открытых систем (Часть 1) / К. А. Бугайский, Д. С. Бирин, Б. О. Дерябин, С. О. Цепенда // Информация и безопасность. – 2022. – Т. 25, № 2. – С. 169–178.*
17. *Котенко И. В. Технологии больших данных для корреляции событий безопасности на основе учета типов связей / И. В. Котенко, А. В. Федорченко, И. Б. Саенко, А. Г. Кушнеревич // Вопросы кибербезопасности. – 2017. – № 5 (24). – С. 2–16. С. 23–32. DOI: 10.21681/2311-3456-2017-5-2-16*
18. *Дойникова Е. В. Совершенствование графов атак для мониторинга кибербезопасности: оперирование неточностями, обработка циклов, отображение инцидентов и автоматический выбор защитных мер / Е. В. Дойникова, И. В. Котенко // Труды СПИИРАН. – 2018. – № 2 (57). – С. 211–240.*
19. *Калашников, А. О. Модель оценки безопасности сложной сети. (часть 1) / А. О. Калашников, К. А. Бугайский // Вопросы кибербезопасности. – 2022. – № 4 (50). – С. 26–38. DOI:10.21681/2311-3456-2022-4-26-38*

ПРОГНОЗИРОВАНИЕ КАТЕГОРИЙ УЯЗВИМОСТЕЙ В КОНФИГУРАЦИЯХ УСТРОЙСТВ С ПОМОЩЬЮ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Левшун Д. С.[1], Веснин Д. В.[2], Котенко И. В.[3]

DOI: 10.21681/2311-3456-2024-3-33-39

Цель исследования: *исследование эффективности модификаций системы двунаправленного обучения трансформеров BERT при решении задачи прогнозирования категорий уязвимостей (CVE) для отдельных элементов (устройств) информационных систем на основе их конфигураций (CPE URIs).*

Методы исследования: *методы обработки естественного языка, кросс-валидация моделей искусственного интеллекта, оптимизация гиперпараметров моделей искусственного интеллекта.*

Полученные результаты: *на основе содержимого открытых баз уязвимостей собран набор данных, устанавливающий взаимосвязи между предобработанными CPE URI и выделенными 24 категориями CVE; исследована эффективность BERT, RoBERTa, XLM-RoBERTa и DeBERTaV3 при решении задачи прогнозирования категорий CVE на основе CPE URI на собранном наборе данных; получена модель BERT, оптимизированная для решения поставленной задачи; произведено сравнение полученного решения с аналогами.*

Научная новизна: *данная работа является одной из первых в прогнозировании уязвимостей устройств на основе их конфигурации, что подчеркивает ее научную значимость и новизну. Более того, это также одна из первых работ, посвященная исследованию BERT для задачи прогнозирования уязвимостей.*

Вклад: *Левшун Д. С., Котенко И. В. – выбор и постановка задачи исследования; Левшун Д. С., Веснин Д. В. – выбор решений, программная реализация и проведение экспериментов; Левшун Д. С., Котенко И. В. – обсуждение результатов экспериментов, анализ полученных результатов.*

Ключевые слова: *информационная безопасность, анализ уязвимостей, BERT, CVE, CPE, CVSS, NVD.*

PREDICTION OF VULNERABILITY CATEGORIES IN CONFIGURATIONS OF DEVICES USING ARTIFICIAL INTELLIGENCE METHODS

Dmitry Levshun[4], Dmitry Vesnin[5], Igor Kotenko[6]

The purpose of the study: *investigation of the effectiveness of BERT modifications in solving the problem of predicting categories of vulnerabilities (CVE) for information system devices based on their configurations (CPE URIs).*

Research methods: *natural language processing methods, cross-validation of artificial intelligence models, optimization of hyperparameters of artificial intelligence models.*

1 Левшун Дмитрий Сергеевич, кандидат технических наук, доктор философии компьютерных наук, старший научный сотрудник Лаборатории проблем компьютерной безопасности, ФГБУН «Санкт-Петербургский Федеральный исследовательский центр Российской академии наук» (СПб ФИЦ РАН), г. Санкт-Петербург, Россия. E-mail: levshun@comsec.spb.ru. ORCID: 0000-0003-1898-6624.

2 Веснин Дмитрий Владимирович, магистр, младший научный сотрудник Лаборатории проблем компьютерной безопасности, ФГБУН «Санкт-Петербургский Федеральный исследовательский центр Российской академии наук» (СПб ФИЦ РАН), г. Санкт-Петербург, Россия. E-mail: vesnin@comsec.spb.ru. ORCID: 0009-0004-8620-2996.

3 Котенко Игорь Витальевич, заслуженный деятель науки РФ, доктор технических наук, профессор, главный научный сотрудник и руководитель лаборатории проблем компьютерной безопасности, ФГБУН «Санкт-Петербургский Федеральный исследовательский центр Российской академии наук» (СПб ФИЦ РАН), г. Санкт-Петербург, Россия. E-mail: ivkote@comsec.spb.ru. ORCID: 0000-0001-6859-7120.

4 Dmitry S. Levshun, Ph.D. (in Tech.), Philosophy Doctor in Computer Science, 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. E-mail: levshun@comsec.spb.ru. ORCID: 0000-0003-1898-6624.

5 Dmitry V. Vesnin, Master Student, 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. E-mail: vesnin@comsec.spb.ru. ORCID: 0009-0004-8620-2996.

6 Igor v. Kotenko, Dr.Sc., Professor, Honored Worker of Science of the Russian Federation, Chief Scientist and Head of Laboratory of Computer Security Problems at St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), St. Petersburg, Russia. E-mail: ivkote@comsec.spb.ru. ORCID: 0000-0001-6859-7120.

Results obtained: *based on the content of open vulnerability databases, we collected a data set that establishes relationships between preprocessed CPE URIs and the identified 24 CVE categories; we investigated the effectiveness of BERT, RoBERTa, XLM-RoBERTa and DeBERTaV3 in solving the problem of predicting CVE categories based on CPE URIs; we trained optimized BERT model to solve the problem of vulnerabilities prediction; we compared the resulting solution with available state-of-the-art.*

Scientific novelty: *this work is one of the first in predicting device vulnerabilities based on their configuration, which emphasizes its scientific significance and novelty. Moreover, it is also one of the first works to explore BERT for the task of vulnerability prediction.*

Contribution: *Levshun D. S., Kotenko I. V. – selection and formulation of the research problem; Levshun D. S., Vesnin D. V. – selection of solutions, software implementation and experiments; Levshun D. S., Kotenko I. V. – discussion of the experimental results, analysis of the results obtained.*

Keywords: *information security, vulnerability analysis, BERT, CVE, CPE, CVSS, NVD.*

Введение

Специалисты по информационной безопасности, ученые и энтузиасты по всему миру усердно работают над обеспечением защиты сетевых систем от вредоносной активности [1]. Данная задача усложняется широким разнообразием угроз и требований безопасности, особенно при защите систем Интернета вещей [2].

Один из популярных подходов к обеспечению безопасности сетевых систем – построение и анализ графов атак [3]. Такие графы позволяют отобразить все доступные пути для злоумышленников через систему, позволяя анализировать как предпосылки, так и последствия атак [4]. В этих графах каждое устройство представляется как узел, а связи между узлами определяются как сетевой политикой, так и потенциалом злоумышленника в компрометации этих устройств. В свою очередь, возможность компрометации устройств определяется наличием уязвимостей в их конфигурации [5].

Самый известный формат описания уязвимостей – CVE (Common Vulnerabilities and Exposures)[7]. CVE хранятся в различных открытых базах данных, наиболее популярной из которых является NVD (National Vulnerability Database)[8]. В NVD содержится почти 200 тысяч CVE, при этом каждая CVE имеет свой уникальный идентификатор, описание, ссылки, уязвимые конфигурации, и т. д.

Уязвимые конфигурации определяются с помощью логических выражений, которые объединяют несколько CPE URIs (Common Platform Enumeration Uniform Resource Identifiers)[9] с помощью логических операторов И и ИЛИ. CPE URI – это структурированная схема именования для всех видов приложений, операционных систем, прошивок и аппаратного обеспечения.

Проблема заключается в том, что конфигурации многих устройств не описаны в открытых базах данных. Это означает, что информация об их уязвимостях не может быть использована при построении графов атак. Таким образом, любое исследование, направленное на прогнозирование уязвимостей в неизвестных конфигурациях, является актуальным. И поскольку каждая уязвимость уникальна, большинство подходов направлены на прогнозирование их метрик.

Метрики уязвимостей описываются в соответствии с стандартом CVSS (Common Vulnerability Scoring System)[10]. В настоящее время наиболее часто используются 2-я (CVSS v2) и 3-я версии (CVSS v3), в то время как 4-я версия была только что представлена и пока не используется в открытых базах данных (CVSS v4). Эти стандарты содержат несколько метрик уязвимостей, 12 представлен в CVSS v2, 9 – CVSS v3.

Для построения графов атак наиболее важны следующие метрики: вектор доступа (access vector, представлен в CVSS v2 и CVSS v3); необходимые привилегии (privileges required, представлены только в CVSS v3,); и получаемые привилегии (obtain privileges, представлены только в CVSS v2 в рамках NVD).

Эти три метрики определяют условия необходимые для успешной эксплуатации уязвимости и ее последствия, а именно, как подключиться к уязвимому устройству (access vector), какие привилегии требуются для эксплуатации уязвимости (privileges required) и какие привилегии получает злоумышленник после эксплуатации (obtained privileges). В предыдущей работе авторов данные метрики были использованы, чтобы разделить все CVE на 24 категории [6].

Ключевым техническим нововведением в области искусственного интеллекта при создании системы BERT (Bidirectional Encoder Representations from Transformers — «двунаправленные представления

7 Официальный веб сайт проекта CVE: https://cve.mitre.org/
8 Официальный веб-сайт базы уязвимостей NVD: https://nvd.nist.gov/
9 Официальный веб-сайт описания конфигураций CPE: https://nvd.nist.gov/products/cpe
10 Официальный веб-сайт описания метрик уязвимостей CVSS: https://www.first.org/cvss/

кодировщика для трансформеров») [7] является применение двунаправленного обучения трансформеров (широко используемой в настоящее время модели с механизмом «внимания») к языковому моделированию.

Целью данной работы является исследование эффективности модификаций BERT для прогнозирования этих категорий в устройствах информационных систем на основе их конфигураций. Данное исследование основано на следующем предположении: CPE URIs, связанные с одинаковыми категориями CVE, более похожи друг на друга, чем CPE URIs, связанные с другими категориями. Таким образом, становится возможным прогнозировать категории CVE для устройств на основе их CPE URIs.

Анализ научной литературы показал, что данная работа является одной из первых в области прогнозирования уязвимостей в устройствах на основе их конфигураций, что подчеркивает ее научную значимость и новизну. Более того, это также одна из первых работ, посвященная исследованию BERT для этой задачи.

Анализ работ

В работе [8] представлен обзор прогнозирования уязвимостей в исходном коде с использованием графовых нейронных сетей (GNN). Авторы сравнили 11 современных методов по архитектуре GNN, по методам представлений графов, наборам данных, точности и F-мере. Важно отметить, что почти все работы использовали различные наборы данных, поэтому сравнить результаты сложно. Например, в представленном сравнении точность варьируется от 58.90 % до 97.40 %, а F-мера – от 36.00 % до 96.11 %. Основной вывод заключается в следующем: существует нехватка реальных наборов данных, поэтому очень важно создать большую базу данных с образцами реального уязвимого исходного кода.

Систематический обзор литературы по обнаружению уязвимостей в программном обеспечении представлен в [9]. Авторы проанализировали 55 исследований, опубликованных с 2015 по 2021 год. Авторы сгруппировали эти исследования в 7 категорий, а именно, нейронные сети, машинное обучение, статический и динамический анализ, клонирование кода, классификация, модели и фреймворки, а также другие для исследований, которые не могут быть включены в эти категории. Было показано, что стратегии машинного обучения широко используются для обнаружения уязвимостей в программном обеспечении, поскольку они позволяют легко анализировать большие объемы данных. Несмотря на разработку многочисленных систем для обнаружения уязвимостей в программном обеспечении, ни одна из них не смогла точно определить конкретный тип обнаруженной уязвимости.

Авторы [10] разработали модель автоматической классификации уязвимостей. Данная модель объединяет TF-IDF (TF – Term Frequency, IDF – Inverse Document Frequency), IG (Information Gain) и глубокие нейронные сети (DNN). TF-IDF используется для определения частоты и важности каждого слова в описании уязвимости, в то время как IG используется для выбора признаков. Затем DNN используется для создания классификатора уязвимостей. Эффективность предложенной модели была проверена на NVD. По сравнению с методами SVM (Support Vector Machine), NB (Naive Bayes) и kNN (k-Nearest Neighbour). Модель авторов показала следующие результаты: аккуратность (accuracy) 87 %, точность (precision) 85 %, полнота (recall) 82 % и F-мера 81 %.

В работе [11] представлен обзор работ по автоматическому обнаружению и прогнозированию уязвимостей программного обеспечения. В этой работе технологии глубокого обучения были разделены на подходы для автоматического обнаружения уязвимостей, исправления программ и прогнозирования дефектов. В качестве основных будущих задач авторы выделили генерацию признаков и параметров, выбор и оценку моделей, а также формирование новых наборов данных.

Анализ работ по прогнозированию уязвимостей программного обеспечения представлен в [12]. Авторы проанализировали 180 исследований, их выводы следующие:

- в уязвимостях программного обеспечения существуют две основные области исследования: прогнозирование уязвимых компонентов программного обеспечения и прогнозирование новых уязвимостей программного обеспечения;
- большинство исследований в области уязвимостей создают собственные наборы данных, собирая информацию из баз данных уязвимостей, содержащих данные о реальном программном обеспечении;
- наблюдается увеличение интереса к моделям глубокого обучения и сдвиг к текстовому представлению исходного кода.

В [13] представлен обзор литературы по подготовке данных для прогнозирования уязвимостей программного обеспечения. Авторы рассмотрели 61 исследование и разработали таксономию подготовки данных для этой задачи. Подготовка данных была разделена на формирование требований (язык программирования, типы уязвимостей, детализация и контекст), сбор данных (реальный мир, синтетический или смешанный код), разметку (предоставленную, сгенерированную или основанную

на шаблонах) и очистку (несущественный код, шум, дублирование).

Анализ современных работ показывает, что прогнозирование уязвимостей с использованием различных типов входных данных находится в настоящее время на стадии активного развития. При этом важно отметить, что прогнозирование категорий уязвимостей на основе конфигураций устройств только начинает исследоваться, что подчеркивает научную значимость и новизну данного направления.

Подход к исследованию

Подход, который был использован для исследования эффективности модификаций BERT, состоит из 4 шагов, начиная от подготовки данных и заканчивая оценкой результатов (рис. 1). Рассмотрим каждый шаг более подробно.

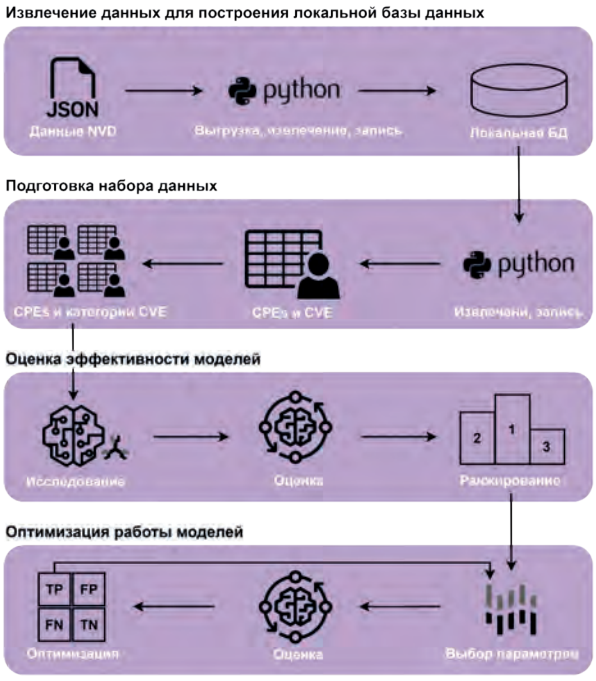


Рис. 1. Подход для исследования эффективности BERT

Шаг 1. Извлечение данных для построения локальной базы данных. На этом шаге данные о CVE выгружаются из файлов в формате JSON, которые доступны в открытой базе уязвимостей NVD. Данные файлы содержат информацию о каждой уязвимости и разделены на три основных типа:

1) файлы с CVE за определенный год;
2) файл с CVE, добавленными за последние 8 дней;
3) файл с CVE, измененными за последние 8 дней.

После загрузки и извлечения всех файлов, необходимо подготовить локальную базу данных для хранения CVE. Для этого в начале разрабатывается структура базы данных, которая затем наполняется данными, извлеченными из выгруженных файлов. В дальнейшем каждые 8 дней данные актуализируются и обновляются с помощью специального скрипта.

Шаг 2. Подготовка набора данных. Внутри базы данных, созданной на предыдущем шаге, данные об уязвимостях организованы в различные таблицы. Следовательно, для извлечения уязвимых конфигураций и их связей с CVE необходимо использовать различные SQL-запросы. Дополнительно, CPE URIs предобрабатываются и связываются с категориями уязвимостей. В осуществленных экспериментах предварительная обработка основана на замене символа «:» пробелом и удалении частей «cpe:2.3:» и «*»:

cpe:2.3:a:gnu:glibc:2.38:*:*:*:*:*:*:* → a gnu glibc 2.38

Итогом данного шага является набор данных для решения задачи с несколькими метками (*multilabel*). Это связано с тем, что различные CPE URI могут быть связаны с несколькими CVE, а CVE могут иметь различные значения метрик CVSS, а значит относиться к разным категориям. Кроме того, удаляются дубликаты из наборов данных.

Шаг 3. Оценка эффективности моделей. На данном шаге мы тестируем модели искусственного интеллекта на наборе данных, полученном на предыдущем шаге. С учетом особенностей BERT, перед подачей CPE URIs, представляющих собой строки текста, к ним применяется токенизация и паддинг.

Важно отметить, что на данном шаге обучение каждой модели осуществляется несколько раз с использованием различных частей набора данных (кросс-валидация). Затем, для каждой метрики эффективности рассчитывается среднее значение, а также среднеквадратичное отклонение. Задачей данного шага является отбор более эффективных моделей искусственного интеллекта для дальнейшей оптимизации их параметров.

Шаг 4. Оптимизация работы моделей. Цель данного шага – подобрать оптимальные гиперпараметры моделей для решаемой задачи, избегая переобучения. В рамках экспериментов оптимизация гиперпараметров осуществлялась с использованием фреймворка Optuna [14]. Итогом работы данного шага является модель, оптимизированная для прогнозирования категорий CVE на основе CPE URI.

Полученный набор данных

В рамках данной работы был создан набор данных для прогнозирования категорий уязвимостей устройств на основе их конфигураций (табл. 1).

Таблица 1

Набор данных для прогнозирования уязвимостей

c_1		c_2		c_3		c_4	
True	False	True	False	True	False	True	False
2261	92779	5723	89317	7526	87514	66684	28356
c_5		c_6		c_7		c_8	
True	False	True	False	True	False	True	False
0	95040	0	95040	0	95040	6	95034
c_9		c_{10}		c_{11}		c_{12}	
True	False	True	False	True	False	True	False
288	94752	15676	79364	523	94517	28266	66774
c_{13}		c_{14}		c_{15}		c_{16}	
True	False	True	False	True	False	True	False
23	95017	8	95032	0	95040	102	94938
c_{17}		c_{18}		c_{19}		c_{20}	
True	False	True	False	True	False	True	False
0	95040	138	94902	32	95008	157	94883
c_{21}		c_{22}		c_{23}		c_{24}	
True	False	True	False	True	False	True	False
102	94938	7353	87687	141	94899	7563	87477

В данном наборе CPE URI связаны с 24 категориями CVE следующим образом:
cpe,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15,c16,c17,c18,c19,c20,c21,c22,c23,c24
a markdown it_project markdown it,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
a oracle jd_edwards_enterpriseone_tools,0,0,0,1,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0
a replit crosis,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0

где cpe представляет собой предобработанные CPE URIs, а c_1 – c_{24} являются бинарными отображениями наличия связи между CPE URI и соответствующей категорией CVE.

Всего в наборе данных 95 040 строк, 26 848 из которых относят одну CPE URI к нескольким категориям уязвимостей (задача с несколькими метками, multilabel). При этом важно отметить, что для категорий c_5, c_6, c_7, c_{15} и c_{17} на данный момент примеры не представлены, что позволяет снизить размерность решаемой задачи с 24 меток до 19. Также отметим, что часть категорий имеет очень малое количество примеров, что затрудняет эффективное решение поставленной задачи ввиду сильной несбалансированности данных.

Полученные результаты

В рамках эксперимента, были протестированы базовые версии следующий моделей – BERT [7], RoBERTa [15], XLM-RoBERTa [16], DeBERTaV3 [17] (табл. 2).

Таблица 2

Параметры протестированных моделей

Модель	Кол-во слоев	Кол-во скрытых слоев	Кол-во параметров, млн.
BERT	12	768	110
RoBERTa	12	768	125
XLM-RoBERTa	12	768	125
DeBERTaV3	12	768	184

На третьем шаге подхода лучшие результаты были продемонстрированы BERT, поэтому только данная модель была передана на оптимизацию гиперпараметров (табл. 3).

Полученное решение способно прогнозировать категории CVE на основе CPE URI с аккуратностью (*accuracy*) равной 0.7226. Насколько нам известно, сравнение данных результатов возможно только с предыдущими работами авторов (табл. 4) [18, 19].

Таблица 3

Результаты оптимизации гиперпараметров

Параметр	Исследованные значения	Оптимальное значение
learning_rate	от 9e-5 до 1e-5 с шагом 1e-5, 9e-4, 8e-4	7e-05
warm_up_epochs	от 0.00 до 1.50 с шагом 0.10	0.40
weight_decay	от 0.00 до 0.05 с шагом 0.01	0.00

Таблица 4

Сравнение полученного решения с аналогами

Подход	Метод	Задача	Модели	Аккуратность
[18]	Прогнозирование категорий CVE	Классификация по одной метке	Random Forest	0.6450
[19]	Прогнозирование метрик CVSS и объединение прогнозов	Классификация по множеству меток	Модификации BERT	0.7382
Предлагаемый	Прогнозирование категорий CVE	Классификация по множеству меток	Модификации BERT	0.7226

Отметим, что хотя не удалось превзойти результаты [19], предлагаемый в данной статье подход также является перспективным. В дальнейшем, при улучшении и доработке использованного набора данных, результаты данного подхода могут превзойти полученные ранее результаты. Это означает, что на данный момент не представляется возможным предположить, какой из подходов покажет большую эффективность.

Заключение

В работе была исследована эффективность таких моделей искусственного интеллекта, как BERT, RoBERTa, XLM-RoBERTa и DeBERTa-v3, для решения задачи прогнозировании категорий уязвимостей в устройствах информационных систем на основе их конфигурации. По итогам экспериментов наилучшие результаты были достигнуты с применением BERT, где аккуратность прогнозов составила 0.7226.

Отметим, что в процессе исследования возник ряд трудностей, связанных с несбалансированностью созданного набора данных. Поэтому в рамках дальнейших исследований, планируется проведение новых экспериментов на расширенном наборе данных, что, как ожидается, позволит улучшить полученные результаты.

Более того, планируется исследовать другие модификации BERT для решения задачи прогнозирования уязвимостей.

Исследование выполнено за счет гранта Российского научного фонда № 22–71–00107, https://rscf.ru/project/22-71-00107/.

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. *Li Y., Huang G., Wang C., Li Y. Analysis framework of network security situational awareness and comparison of implementation methods // EURASIP Journal on Wireless Communications and Networking. 2019. Vol. 2019. P. 1–32. DOI: 10.1186/s13638-019-1506-1.*
2. *Израилов К. Е., Левшун Д. С., Чечулин А. А. Модель классификации уязвимостей интерфейсов транспортной инфраструктуры «умного города» // Системы управления, связи и безопасности. 2021. №. 5. С. 199–223. DOI: 10.24412/2410-9916-2021-5-199-223.*
3. *Lallie H. S., Debattista K., Bal J. A review of attack graph and attack tree visual syntax in cyber security // Computer Science Review. 2020. Vol. 35. P: 100219. DOI: 10.1016/j.cosrev.2019.100219.*
4. *Федорченко Е. В., Котенко И. В., Федорченко А. В., Новикова Е. С., Саенко И. Б. Оценивание защищенности информационных систем на основе графовой модели эксплойтов // Вопросы кибербезопасности. 2023. № 3 (55). С.23-36. DOI:10.21681/2311-3456-2023-3-23-36.*
5. *Kotenko I., Izrailov K., Buinevich M., Saenko I., Shorey R. Modeling the Development of Energy Network Software, Taking into Account the Detection and Elimination of Vulnerabilities // Energies. 2023. Volume 16, Issue 13, 5111. P.1-40. https://doi.org/10.3390/en16135111.*
6. *Levshun D., Chechulin A. Vulnerability Categorization for Fast Multistep Attack Modelling // Proceedings of the 33rd Conference of the Open Innovations Association FRUCT. May 24-26, Zilina, Slovakia. 2023. P. 169-175. DOI: 10.23919/FRUCT58615.2023.10143048.*

7. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol.1. 2019. P.4171–4186. DOI:10.18653/v1/N19-1423.
8. Katsadouros E., Patrikakis C. A Survey on Vulnerability Prediction using GNNs // Proceedings of the 26th Pan-Hellenic Conference on Informatics. 2022. P. 38-43. DOI: 10.1145/3575879.3575964.
9. Eberendu A. C., Udegbe V. I., Ezennorom E. O., Ibegbulam A. C., Chinebu T. I. A systematic literature review of software vulnerability detection // European Journal of Computer Science and Information Technology. 2022. Vol. 10. No. 1. P. 23–37. DOI: 10.37745/ejcsit.2013.
10. Huang G., Li Y., Wang Q., Ren J., Cheng Y., Zhao, X. Automatic classification method for software vulnerability based on deep neural network // IEEE Access. 2019. Vol. 7. P. 28291-28298. DOI: 10.1109/ACCESS.2019.2900462.
11. Shen Z., Chen S. A survey of automatic software vulnerability detection, program repair, and defect prediction techniques // Security and Communication Networks. 2020. Vol. 2020. P. 1–16. DOI: 10.1155/2020/8858010.
12. Kalouptsoglou I., Kalouptsoglou I., Siavvas M., Ampatzoglou A., Kehagias D., Chatzigeorgiou A. Software vulnerability prediction: A systematic mapping study // Information and Software Technology. 2023. P. 107303. DOI: 10.1016/j.infsof.2023.107303.
13. Croft R., Xie Y., Babar M. A. Data preparation for software vulnerability prediction: A systematic literature review // IEEE Transactions on Software Engineering. 2022. Vol. 49. No. 3. P. 1044-1063. DOI: 10.1109/TSE.2022.3171202.
14. Akiba T., Sano S., Toshihiko Y., Ohta T., Koyama M. Optuna: A next generation hyperparameter optimization framework // Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019. P. 2623–2631. DOI: 10.1145/3292500.3330701.
15. Liu Z., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. A robustly optimized BERT pre-training approach with post-training // Proceedings of the China National Conference on Chinese Computational Linguistics. Cham: Springer International Publishing, 2021. P. 471–484. DOI: 10.48550/arXiv.1907.11692.
16. Conneau A., Chaudhary V., Wenzek G., Guzman F., Grave E., Ott M., Zettlemoyer L., Stoyanov V. Unsupervised Cross-lingual Representation Learning at Scale // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics. 2020. DOI: 10.48550/arXiv.1911.02116.
17. He P., Gao J., Chen W. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing // Proceedings of the Eleventh International Conference on Learning Representations. 2022. DOI: 10.48550/arXiv.2111.09543.
18. Levshun D. Comparative analysis of machine learning methods in vulnerability categories prediction based on configuration similarity // Proceedings of the 16th International Symposium on Intelligent Distributed Computing (IDC-2023). September 13–15, Hamburg, Germany. 2023. P. 231–242.
19. Levshun D., Vesnin D. Exploring BERT for Predicting Vulnerability Categories in Device Configurations // Proceedings of the 10th International Conference on Information Systems Security and Privacy (ICISSP 2024). February 26–28, Rome, Italy. 2024. P. 452–461. DOI: 10.5220/0012471800003648.

ПРОБЛЕМЫ ОЦЕНКИ ДОВЕРИЯ К ПРОЦЕССАМ АУДИТА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Иванов А. В.[1], Огнев И. А.[2]

DOI: 10.21681/2311-3456-2024-3-40-50

Цель исследования: *формирование алгоритма оценки доверия к процессу аудита информационной безопасности, состоящего из последовательного многоэтапного анализа доказательств доверия по иерархической модели «объект-критерии-метрики».*

Методы исследования *базируются на анализе отечественного и зарубежного нормативно-правового поля, научных публикаций, а также на применении функции желательности Харрингтона.*

Результат: *был сформирован алгоритм проведения оценки доверия, состоящий из последовательного многоэтапного анализа доказательств доверия по иерархической модели «объект-критерии-метрики». В соответствии с данной иерархической моделью метрики вычисляются на основе анализа доказательств доверия, критерии – на основе значений метрик, а уровень доверия – на основе значений критериев. Были определены метрики и критерии оценки доверия. Расчет доверия к процессу аудита информационной безопасности базируется на функции желательности Харрингтона и ГОСТ Р 57580.2–2018. В данном случае метрики, как числовой результат оценки доказательств доверия, выступают частными признаками желательности, критерии – как частные функции желательности, а уровень доверия к процессу аудита информационной безопасности – как обобщенная функция желательности.*

Полученный алгоритм оценки доверия к процессам аудита информационной безопасности будет интегрирован в общий алгоритм оценки доверия к субъектам информационного обмена, который включает в себя анализ ряда процессов информационной безопасности, одним из которых является аудит.

Научная новизна *заключается в предложении динамического метода контроля процесса аудита информационной безопасности, основанного на анализе объективных свидетельств и подлежавшего автоматизации. Оценка доверия, как динамическая мера контроля процессов информационной безопасности, призвана минимизировать трудо- и времязатраты при контроле процессов информационной безопасности.*

Ключевые слова: *доверие, оценка доверия, оценка соответствия, доверие к аудиту, оценка процессов, доверенное взаимодействие, информационная безопасность, кибербезопасность.*

PROBLEMS OF ASSESSING TRUST IN INFORMATION SECURITY AUDIT PROCESSES

Ivanov A. V.[3], Ognev I. A.[4]

The purpose of the study: *the formation of an algorithm for assessing trust in the information security audit process, consisting of a sequential multi-stage analysis of evidence of trust according to the hierarchical model «object-criteria-metrics».*

The research methods *are based on the analysis of the domestic and foreign regulatory framework, scientific publications, as well as on the application of Harrington's desirability function.*

Result: *an algorithm for assessing trust was formed, consisting of a sequential multi-stage analysis of evidence of trust according to the hierarchical model «object-criteria-metrics». In accordance with this hierarchical*

1 Иванов Андрей Валерьевич, кандидат технических наук., доцент, заведующий кафедрой защиты информации, Новосибирский государственный технический университет (НГТУ), Новосибирск, РФ. E mail: andrej.ivanov@corp.nstu.ru

2 Огнев Игорь Александрович, аспирант, ассистент кафедры защиты информации, Новосибирский государственный технический университет (НГТУ), Новосибирск, РФ. E-mail: i.ognev.2016@corp.nstu.ru

3 Andrey V. Ivanov, Ph.D., Associate Professor, Head of the Department of Information Security, Novosibirsk State Technical University (NSTU), Novosibirsk, Russian Federation. E-mail: andrej.ivanov@corp.nstu.ru

4 Igor A. Ognev, graduate student, assistant at the Department of Information Security, Novosibirsk State Technical University (NSTU), Novosibirsk, Russian Federation. E-mail: i.ognev.2016@corp.nstu.ru

model, metrics are calculated based on the analysis of evidence of trust, criteria are calculated based on the values of the metrics, and the level of trust is calculated based on the values of the criteria. Metrics and criteria for assessing trust were defined. The calculation of trust in the information security audit process is based on the Harrington desirability function and GOST R 57580.2–2018. In this case, metrics, as a numerical result of assessing evidence of trust, act as partial signs of desirability, criteria – as partial functions of desirability, and the level of confidence in the information security audit process – as a generalized function of desirability.

The resulting algorithm for assessing trust in information security audit processes will be integrated into the general algorithm for assessing trust in subjects of information exchange, which includes an analysis of a number of information security processes, one of which is audit.

***The scientific novelty** lies in the proposal of a dynamic method for monitoring the information security audit process, based on the analysis of objective evidence and subject to automation. Trust assessment, as a dynamic measure of control of information security processes, is designed to minimize labor and time costs when monitoring information security processes.*

***Keywords:** trust, methodology, conformity assessment, audit trust, process assessment, trusted interaction, information security, cybersecurity.*

Введение

Текущая ситуация в сфере информационной безопасности явно указывает на то, что организациям необходимо постоянно повышать свой уровень защищенности, как от внешних, так и от внутренних угроз. Positive Technologies, ведущая компания-разработчик в сфере информационной безопасности[5], основываясь на статистике, полученной от более чем 2300 организаций в России, отмечает постоянный рост количества киберугроз. В 2020 г. по сравнению с 2019 г. рост составил 51%, при этом 70% атак носили целенаправленный характер[6]; в 2021 г. по сравнению с 2020 г. рост составил 6,5%, при этом доля целевых атак возросла на 4% и составила 74% от общего количества атак[7]. Эксперты «Лаборатории Касперского» отмечают резкий рост количества сложных кибератак в 4 раза в первом квартале 2022 г. по сравнению с аналогичным периодом в 2021 г[8]. Основой для построения сложных атак являются небольшие бреши в системе безопасности организации [1]. Для планирования векторов проведения атаки злоумышленники используют результаты нелегитимного исследования, оценивая уязвимости – как известные, так и еще необнародованные, так называемые уязвимости нулевого дня (0-day). Из-за технической сложности обнаружения таких брешей стандартными средствами защиты информации необходимо выстроить непрерывный процесс анализа и контроля систем защиты информации и процессов информационной безопасности.

В современных условиях функционирования информационных систем вопрос о подтверждении состояния защищенности субъектов информационного обмена становится актуальным. Под субъектом информационного обмена понимается юридическое лицо или орган государственной власти, которому на праве владения, аренды или ином законном основании принадлежат информационные системы (государственные информационные системы, информационные системы персональных данных, автоматизированные системы управления технологическими процессами, объекты критической информационной инфраструктуры). Существующие статические методы[9] [2, 3] оценки защищенности субъектов информационного обмена не дают актуальную информацию о состоянии системы защиты, что приводит к необходимости использовать динамические методы оценки состояния систем защиты информации. Разрабатываемая технология оценки уровня доверия к субъектам информационного обмена [4, 5] позволит оперативно оценивать состояние систем безопасности участников информационного обмена и реагировать на изменение этого состояния. Данная технология позволит обеспечить высокую скорость реагирования на нарушения состояния безопасности, возникающие при взаимодействии участников информационного обмена, и своевременное принятие необходимых мер по их устранению.

5 О компании // Positive Technologies [Электроный ресурс]. 2022 – URL: https://www.ptsecurity.com/ru-ru/about/ (дата обращения: 12.02.2023).

6 Актуальные киберугрозы: итоги 2020 года // Positive Technologies [Электроный ресурс]. 2021 – URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2020 (дата обращения: 12.02.2023).

7 Positive Technologies: число кибератак в 2021 году выросло на 6,5% // Positive Technologies [Электронный ресурс]. 2022 – URL: https://www.ptsecurity.com/ru-ru/about/news/chislo-kiberatak-v-2021-godu-vyroslo-na-6-5-procentov/ (дата обращения: 12.02.2023)

8 «Лаборатория Касперского»: количество киберинцидентов в российских компаниях увеличилось в 4 раза // Лаборатория Касперского [Электронный ресурс]. 2022 – URL: https://www.kaspersky.ru/about/press-releases/2022_laboratoriya-kasperskogo-kolichestvo-kiberincidentov-v-rossijskih-kompaniyah-uvelichilos-v-4-raza (дата обращения: 12.02.2023).

9 ФСТЭК России «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» от 11 февраля 2013 г. № 17 // https://fstec.ru/dokumenty/vse-dokumenty/prikazy/prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17. – 2013 г. – с изм. и допол. в ред. от Приказов ФСТЭК России от 15.02.2017 № 27, от 28.05.2019 № 106.

Технология оценки доверия к субъектам информационного обмена включает в себя комплексный анализ ряда процессов информационной безопасности. В рамках настоящего исследования разрабатывается алгоритм проведения оценки доверия к процессу аудита информационных систем, включая экспертный (экспертно-аналитический) и активный (инструментальный) аудит. Будут рассмотрены вопросы трактовки термина доверие применительно к процессу оценки доверия, описания процесса аудита, как объекта оценки доверия, формирования алгоритма оценки доверия к процессу аудита информационной безопасности, включая описание выходных данных процесса, логику анализа и обработки входных данных, описание выходных данных для принятия решении о возможности построения доверенного взаимодействия с контрагентами.

1. Вопросы оценки доверия

В настоящее время термин доверия не имеет единой трактовки, что является одной из первостепенных проблем исследования. В различных источниках (отечественное и зарубежное нормативно-правовое поле, отечественные и зарубежные научные публикации) можно найти трактовку доверия в качестве:

1) доверия к техническим или программным средствам:
   a. как процесс оценки соответствия средств защиты информации требованиям по безопасности, включающих требования к разработке и производству средства, к проведению испытаний средства, к поддержке безопасности средства[10,11];
   b. как доверие к устройствам IoT [6] или узлам сетей типа Bode Area Network [7];
2) доверия к субъекту информационных систем (пользователь, программа и т.д.) – архитектура нулевого доверия, смысл которой заключается в формировании правил идентификации и аутентификации пользователей на основе отсутствия неявного доверия к активам и учетным записям организации, основанного на их физическом или сетевом местоположении, а также на основе владельца активов[12] [8, 9, 10];
3) доверия к информации – алгоритмы оценки доверия к информации на основе происхождения информации и (или) источника информации [11, 12, 13].

В рамках настоящего исследования будем трактовать процесс оценки доверия как процесс оценки соответствия субъектов информационного обмена требованиям доверия, путем оценки процессов информационной безопасности, которые будут рассмотрены далее по тексту.

Цель оценки уровня доверия заключается в создании объективных доказательств, которые позволяют в произвольный момент времени убедиться в невозможности реализации неприемлемых рисков злоумышленником. Это также включает в себя риски, которые оператор или владелец информационной системы принял без учета информационного взаимодействия. Такая оценка помогает обеспечить безопасность информационных систем и минимизировать возможные угрозы.

Например, аттестат соответствия[13], как статическая мера контроля, свидетельствует о том, что система защиты информации правильно организована и соответствует всем необходимы требованиям по защите информации, но процесс аттестации проводится перед вводом системы защиты информации в эксплуатацию, и аттестат действует бессрочно. Исходя из этого аттестат не дает уверенности в том, система защиты информации правильно и исправно функционирует спустя некоторое время (например, через 3 месяца или через год). Для формирования уверенности контрагентов в том, что система защиты информации и процессы информационной безопасности правильно функционируют, можно использовать динамический метод контроля, который за короткий промежуток времени на основе объективных показателей даст заключение о состоянии информационной безопасности контрагента.

Общая оценка уровня доверия к субъекту информационного обмена состоит из оценки ряда внутренних процессов и процедур информационной безопасности субъекта информационного обмена:

1) системы управления рисками;
2) системы управления угрозами и уязвимостями;
3) процессов аудита;
4) системы управления информационной безопасностью;
5) процедур эксплуатации средств защиты информации;
6) процедур создания системы управления информационной безопасностью;
7) информационных технологий.

В данной работе объектом исследования является процесс аудита. Под процессом аудита будем понимать процесс получения свидетельств о состоянии информационной безопасности объекта аудита

10 Требования по безопасности информации, устанавливающие уровни доверия к средствам технической защиты информации и средствам обеспечения безопасности информационных технологий, утвержденным приказом ФСТЭК России от 2 июня 2020 г. № 76.

11 ГОСТ Р ИСО/МЭК 15408–3–2008 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Компоненты доверия к безопасности».

12 NIST SP 800–207 Zero Trust Architecture

13 ФСТЭК России «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» от 11 февраля 2013 г. № 17 // https://fstec.ru/dokumenty/vse-dokumenty/prikazy/prikaz-fstek-rossii-ot-11-fevralya-2013-g-n-17. - 2013 г. - с изм. и допол. в ред. от Приказов ФСТЭК России от 15.02.2017 № 27, от 28.05.2019 № 106.

и процесс оценки свидетельств с целью установления степени соответствия критериям аудита. В данное понятие включаются экспертный, нормативный аудит, а также технический аудит (оценка защищенности).

2. Оценка доверия к процессу аудита информационной безопасности

Анализ научных и нормативных источников по тематике аудита информационной безопасности показал, что обобщенно процесс аудита представляет собой ряд последовательных шагов[14,15] [14]:

1. Формирование команды аудита;
2. Предварительное обследование объекта аудита;
3. Формирование программы аудита, включающую определение методов аудита и критериев аудита;
4. Обследование объекта аудита и сбор свидетельств аудита;
5. Оценка свидетельств аудита полностью или выборочно на соответствие критериям аудита;
6. Формирование итогового заключения аудита с указанием замечаний.

Оценку уровня доверия к аудиту предлагается проводить в соответствии с линейной моделью оценки «критерий–метрика», которая основана на модели «фактор–критерий–метрика»[16] (рис. 1).

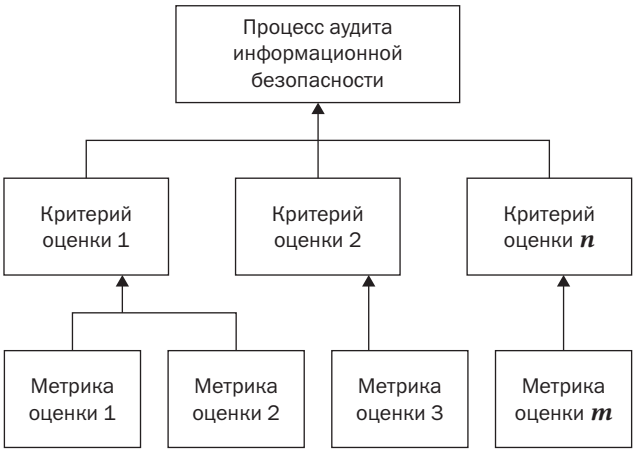


Рис. 1. Оценка процессов по модели «критерий-метрика»

Под объектом оценки будем понимать процесс аудита, под критериями – оцениваемые свойства процесса аудита, под метриками – конкретные свидетельства аудита, подлежащие оценке.

Оценка уровня доверия к процессу аудита информационной безопасности заключается в ряде последовательных шагов (рис. 2):

1) формирование и передача доказательств доверия – самостоятельный сбор доказательств доверия к процессу аудита информационной безопасности субъектом информационного обмена;
2) анализ доказательств доверия на предмет соответствия требованиям доверия – расчет значений метрик доверия на основе анализа доказательств доверия;
3) оценка свойств процесса аудита информационной безопасности – расчет значений критериев доверия на основе значений метрик доверия;
4) расчет значения уровня доверия к процессу аудита информационной безопасности субъекта информационного обмена на основе значений критериев доверия.

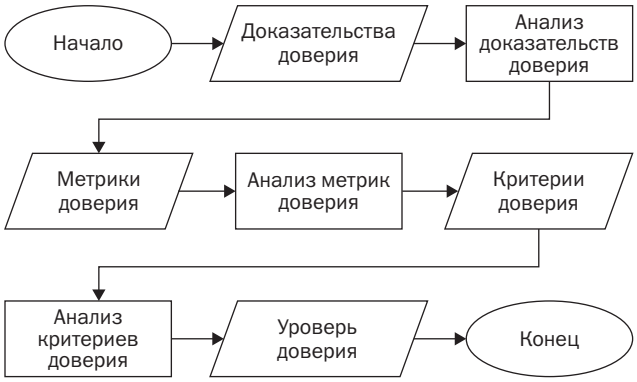


Рис. 2. Оценка доверия к процессу аудита

В данном случае доказательства доверия являются входными данными процесса оценки доверия к процессу аудита информационной безопасности, вычисление метрик, критериев доверия и уровня доверия являются логикой обработки доказательств доверия, уровень доверия – является выходным значением процесса оценки доверия к процессу аудита информационной безопасности. Более подробно доказательства доверия, критерии и метрики доверия рассмотрим далее.

Данный алгоритм проведения процесса оценки уровня доверия базируется на процессе проведения оценки соответствия по ряду ГОСТов.[17,18,19]

2.1. Доказательства доверия

При проведении оценки уровня доверия к аудиту субъекта информационного обмена необходимо осуществить сбор доказательств доверия (таблица 1).

14 ГОСТ Р ИСО/МЭК 27007-2014 «Информационная технология. Методы и средства обеспечения безопасности. Руководства по аудиту систем менеджмента информационной безопасности»

15 ГОСТ Р ИСО 19011–2021. Оценка соответствия. Руководящие указания по проведению аудита систем менеджмента качества. – М.: Стандартинформ, 2021. – 35 с

16 ГОСТ 28195–89 Оценка качества программных средств. Общие положения.

17 ГОСТ Р 57580.2–2018 «Безопасность финансовых (банковских) операций. Защита информации финансовых организаций. Базовый состав организационных и технических мер. Методика оценки соответствия».

18 ГОСТ Р ИСО/МЭК 27007–2014 «Информационная технология. Методы и средства обеспечения безопасности. Руководства по аудиту систем менеджмента информационной безопасности».

19 ГОСТ Р ИСО 19011–2021 «Оценка соответствия. Руководящие указания по проведению аудита систем менеджмента».

Таблица 1

Доказательства доверия

№ п/п	Наименование доказательства	Обозначение
1	Акт приема-передачи (на средства контроля защищенности)	P_1
2	Программа аудита	P_2
3	Заключение аудита	P_3
4	Приказ о формировании комиссии (отдела) внутреннего аудита	P_4
	Договор (соглашение) о проведении внешнего аудита	
5	План мероприятий по обеспечению безопасности информации	P_5
6	План мероприятий по актуализации состава информационной системы и (или) подсистемы безопасности	P_6
7	Наличие в плане мероприятий по обеспечению безопасности информации сведений о проведении аудита	P_7
8	Отчеты об устранении замечаний из заключений аудита	P_8
9	Сведения о составе информационной системы и подсистемы защиты информации	P_9
10	Сведения о делах Арбитражных судов в отношении поставщиков услуг аудита в связи с невыполнением или недобросовестным выполнением обязательств (при наличии поставщика услуг аудита)	P_{10}
11	Выписка из личных дел сотрудников, входящих в состав комиссии (отдела) аудита	P_{11}
12	Сертификаты о повышении квалификации сотрудников, входящих в состав комиссии (отдела) аудита	P_{12}
13	Лицензия ФСТЭК России на предоставление услуг по контролю защищенности конфиденциальной информации от несанкционированного доступа и ее модификации в средствах и системах информатизации	P_{13}

Эти доказательства субъект информационного обмена собирает и передает самостоятельно. Набор доказательств сформирован на основе нормативного сопровождения процесса аудита информационной безопасности[20,21,22], а также на основе возможности подтверждения ряда фактов, свидетельствующих о качестве и добросовестности подхода к организации и проведению аудита информационной безопасности [15].

Каждое доказательство имеет свое уникальное обозначение для формирования формул расчета метрик доверия на основе анализа доказательств доверия. Данные вопросы будут рассмотрены далее.

Точные наименования документов могут отличаться от указанных в таблице 1 в соответствии с устоявшимися в субъектах информационного обмена наименованиями. Обозначения доказательств P_n введены для их упрощенного обозначения далее по тексту.

Оценка соответствия доказательств доверия требованиям доверия осуществляется в виде расчета метрик доверия. Числовое значение соответствия доказательства доверия требованиям доверия является метрикой доверия.

2.2. Расчет метрик доверия

Оценка доказательств доверия (расчет метрик доверия) необходима для получения численных показателей – метрик доверия. Анализ доказательств доверия заключается в выявлении ряда фактов [16, 17]:

1. Факт наличия доказательства доверия или факт наличия иных признаков доверия, содержащихся в доказательствах доверия;
2. Отношения количественных показателей какого-либо из свойств системы, полученных из доказательств доверия, к общему номинальному значению уникальному для каждого изучаемого объекта (например, отношение числа процессов или систем, попавших под аудит, к общему числу процессов или систем).

Метрики сгруппированы в 4 критерия доверия, каждый из которых является каким-либо свойством

20 ГОСТ Р ИСО/МЭК 27007-2014 «Информационная технология. Методы и средства обеспечения безопасности. Руководства по аудиту систем менеджмента информационной безопасности»

21 ГОСТ Р ИСО 19011–2021. Оценка соответствия. Руководящие указания по проведению аудита систем менеджмента качества. – М.: Стандартинформ, 2021. – 35 с

22 Макаренко С. И. Аудит информационной безопасности: основные этапы, концептуальные основы, классификация мероприятий // Системы управления, связи и безопасности. — 2018. — № 1. — С. 1–29.

оцениваемого процесса информационной безопасности – процесса аудита информационной безопасности:

1) доверие к полноте аудита;
2) доверие к качеству аудита;
3) доверие к своевременности аудита;
4) доверие к поставщику услуг аудита.

Значения метрик лежат в промежутке [0, 1], где 0 – самая низкая оценка, а 1 – самая высокая. Более конкретные формулы приведены далее.

Для обозначения критериев и метрик примем обозначения M_n для критериев и M_{nm} для метрик. Соответственно критерий доверия к полноте аудита будет иметь обозначение M_1, а метрики данного критерия – M_{1m}. Критерий доверия к качеству аудита и его метрики будут иметь обозначения M_2 и M_{2m} соответственно, критерий доверия к своевременности и его метрики – M_3 и M_{3m}, критерий доверия к поставщику услуг и его критерии – M_4 и M_{4m}.

2.2.1. Оценка метрик доверия к полноте аудита

В качестве исходных данных для вычисления метрик для оценки доверия к полноте процесса аудита информационной безопасности предлагается использовать следующие доказательства доверия:

1) акт приема – передачи (на средства контроля защищенности);
2) программа аудита;
3) заключение аудита;
4) приказ о формировании комиссии (отдела) внутреннего аудита или договор (соглашение) о проведении внешнего аудита;
5) сведения о составе информационной системы и подсистемы защиты информации.

Состав метрик оценки доверия к полноте аудита (таблица 2) и формулы для их вычисления приведены ниже.

Таблица 2

Метрики оценки полноты аудита

Номер метрики	Наименование
M_{11}	Наличие программных средств контроля защищенности
M_{12}	Наличие программы аудита
M_{13}	Наличие заключения аудита
M_{14}	Наличие временной или постоянной группы аудита
M_{15}	Отношение процессов/систем, попадающих под аудит к общему числу процессов/систем
M_{16}	Наличие замечаний в заключении аудита

Метрики вычисляются по следующим формулам:

$$M_{11} = \begin{cases} 0, \nexists P_1 \\ 1, \exists P_1 \end{cases} \quad (1)$$

$$M_{12} = \begin{cases} 0, \nexists P_2 \\ 1, \exists P_2 \end{cases} \quad (2)$$

$$M_{13} = \begin{cases} 0, \nexists P_3 \\ 1, \exists P_3 \end{cases} \quad (3)$$

$$M_{14} = \begin{cases} 0, \nexists P_4 \\ 1, \exists P_4 \end{cases} \quad (4)$$

$$M_{15} = \begin{cases} 0, \nexists P_2 \vee \nexists P_9 \\ \dfrac{N_{ауд}}{N_{общ}}, \exists P_4 \wedge \exists P_9 \end{cases} \quad (5)$$

где $N_{ауд}$ – количество процессов/систем, в отношении которых проводится аудит, $N_{общ}$ – общее количество процессов/систем.

$$M_{15} = \begin{cases} 0, \nexists Z_{крит} \wedge \nexists Z_{нс} \\ 0.5, \nexists Z_{крит} \wedge \exists Z_{нс} \\ 1, \exists Z_{крит} \end{cases} \quad (6)$$

где $Z_{крит}$ – критические замечания, выявленные при аудите и требующие устранения, $Z_{нс}$ – несущественные замечания, выявленные при аудите и требующие устранения, при этом $(Z_{крит} \cap Z_{нс}) \subseteq P_3$ [15].

По результатам оценки метрик полноты аудита информационной безопасности можно, во-первых, выявить недостатки процесса аудита информационной безопасности, а во-вторых, рассчитать числовое значение доверия к полноте аудита. Данные расчеты приведены далее в пп. 2.3.

Далее рассмотрим вопросы расчета метрик доверия к качеству процесса аудита информационной безопасности.

2.2.2. Оценка метрик доверия к качеству аудита

В качестве исходных данных для вычисления метрик оценки доверия к качеству процесса аудита информационной безопасности предлагается использовать следующие доказательства доверия:

1) акт приема-передачи (на средства контроля защищенности);
2) программа аудита;
3) заключение аудита;
4) план мероприятий по обеспечению безопасности информации;
5) отчеты об устранении замечаний из заключений аудита.

Состав метрик оценки доверия к качеству аудита (таблица 3) и формулы их вычисления приведены ниже.

Таблица 3

Метрики оценки качества аудита

Номер метрики	Наименование
M_{21}	Наличие программных средств контроля защищенности
M_{22}	Соответствие аудита одной из методологий проведения аудита или оценки защищенности
M_{23}	Наличие замечаний в заключении аудита
M_{24}	Наличие отчета о проведении мероприятий по устранению замечаний в заключении аудита
M_{25}	Отношение времени, затраченного на устранение замечаний, к допустимому времени устранения замечаний

Аналогично метрикам оценки доверия к полноте процесса аудита информационной безопасности проводится расчет метрик доверия к качеству аудита: метрика M_{21} рассчитывается по формуле (1), метрика M_{22} – по формуле (2), метрика M_{23} – по формуле (6). Расчет остальных метрик приведен ниже.

$$M_{24} = \begin{cases} 0, \nexists P_5 \\ 1, \exists P_5 \end{cases} \quad (7)$$

$$M_{25} = \begin{cases} 0, \nexists P_8 \vee t_y > 1.5 t_д \\ 0.5, t_y \le 1.5 t_д \\ 1, t_y \le t_д \end{cases} \quad (8)$$

где t_y – время, затраченное на исправление замечаний, выявленных при аудите, $t_д$ – допустимое время устранения замечаний, устанавливаемое регулятором, командой проведения аудита или планом по устранению замечаний.

Расчет числового значения доверия к качеству процесса аудита информационной безопасности приведен далее в пп. 2.3.

Далее рассмотрим вопросы расчета метрик доверия к своевременности процесса аудита информационной безопасности.

2.2.3. Оценка метрик доверия к своевременности аудита

В качестве исходных данных для вычисления метрик оценки доверия к своевременности процесса аудита информационной безопасности предлагается использовать следующие доказательства доверия:

1) план мероприятий по актуализации состава информационной системы и (или) подсистемы безопасности;
2) наличие сведений о проведении аудита в плане мероприятий по обеспечению безопасности информации;
3) сведения о составе информационной системы и подсистемы защиты информации.

Состав метрик оценки доверия к своевременности аудита (таблица 4) и формулы их вычисления приведены ниже.

Таблица 4

Метрики оценки своевременности аудита

Номер метрики	Наименование
M_{31}	Наличие фактов проведения инвентаризации компонентов ИС и периодичность проведения инвентаризации
M_{32}	Наличие фактов проведения аудита информационной безопасности и периодичность проведения аудита
M_{33}	Наличие факта обновления сведений об инвентаризации при изменениях в ИС

Приведенные метрики вычисляются по следующим формулам:

$$M_{31} = \begin{cases} 0, \nexists P_6 \vee t_u > 5 \\ \dfrac{0.5}{T_u}, 0 < T_u \le 5 \end{cases} \quad (9)$$

где T_u – периодичность проведение инвентаризации в годах (0,5–5)

$$M_{32} = \begin{cases} 0, \nexists P_7 \vee t_A > 5 \\ \dfrac{0.5}{T_A}, 0 < T_A \le 5 \end{cases} \quad (10)$$

где ТА – периодичность проведение аудита в годах (0,5; 5)

$$M_{24} = \begin{cases} 0, \nexists P_9 \\ 1, \exists P_9 \end{cases} \quad (11)$$

Расчет числового значения доверия к своевременности процесса аудита информационной безопасности приведен далее в пп. 2.3.

Далее рассмотрим расчет метрик доверия к поставщику услуг процесса аудита информационной безопасности.

2.2.4. Оценка доверия к поставщику услуг аудита

В качестве исходных данных для вычисления метрик оценки доверия к поставщику услуг процесса аудита информационной безопасности предлагается использовать следующие доказательства доверия:

1) сведения о делах арбитражных судов в отношении поставщиков услуг аудита в связи с невыполнением или недобросовестным выполнением обязательств (при наличии внешнего поставщика услуг аудита);
2) выписка из личных дел сотрудников, входящих в состав комиссии (отдела) аудита;
3) сертификаты о повышении квалификации сотрудников, входящих в состав комиссии (отдела) аудита;

4) лицензия ФСТЭК России на предоставление услуг по контролю защищенности конфиденциальной информации от несанкционированного доступа и ее модификации в средствах и системах информатизации;
5) акт приема – передачи (на средства контроля защищенности);
6) приказ о формировании комиссии (отдела) внутреннего аудита или договор (соглашение) о проведении внешнего аудита.

Состав метрик оценки доверия к поставщику услуг аудита (таблица 5) и формулы их вычисления приведены ниже.

Таблица 5

Метрики оценки поставщика услуг аудита

Номер метрики	Наименование
M_{41}	Наличие фактов недобросовестности поставщика услуг аудита (только при привлечении внешнего поставщика услуг)
M_{42}	Подтверждение опыта работы специалистов команды аудита в области аудита информационной безопасности
M_{43}	Наличие фактов повышения квалификации специалистов команды аудита в области аудита информационной безопасности и периодичности прохождения курсов повышения квалификации
M_{44}	Наличие программных средств контроля защищенности
M_{45}	Соответствие процедур аудита и оценки защищенности одной из методологий проведения аудита/оценки защищенности
M_{46}	Наличие лицензий на выполнение работ по аудиту/оценке защищенности (пп. б п. 4 ПП РФ 79) (только при привлечении внешнего поставщика услуг)

Аналогично метрикам оценки доверия к полноте процесса аудита информационной безопасности проводится расчет метрик доверия к качеству аудита: метрика M_{44} рассчитывается по формуле (1), метрика M_{45} – по формуле (4). Расчет остальных метрик приведен ниже.

$$M_{41} = \begin{cases} \frac{1}{N_c}, N_c \geq 1 \\ 1, \nexists P_{10} \vee N_c = 0 \end{cases} \quad (12)$$

где N_c – количество удовлетворительных исков о неисполнении или ненадлежащем исполнении обязательств в Арбитражном суде за последние 3 года, в которых поставщик услуг ответчик.

$$M_{42} = \begin{cases} 0, \nexists P_{11} \vee O < 3 \\ 1, O \geq 3 \end{cases} \quad (13)$$

где O – стаж работы в годах специалистов, входящих в команду аудита, в области аудита или оценки защищенности.

$$M_{43} = \begin{cases} 0, \nexists P_{12} \vee T\kappa < 3 \\ 1, T\kappa \geq 3 \end{cases} \quad (14)$$

где $T\kappa$ – периодичность проведения повышения квалификации в годах.

$$M_{46} = \begin{cases} 0, \nexists P_{13} \\ 1, \exists P_{13} \end{cases} \quad (15)$$

По результатам оценки метрик доверия к поставщику услуг аудита информационной безопасности можно, во-первых, выявить недостатки в команде аудита информационной безопасности, а во-вторых, рассчитать числовое значение доверия к поставщику услуг аудита.

Далее перейдем к расчету критериев доверия к качеству процесса аудита информационной безопасности, а именно к расчету доверия к полноте, качеству, своевременности аудита и доверия к поставщику услуг, на основе значений метрик доверия, описанных ранее в пп. 2.2.1–2.2.4.

2.3. Расчет критериев доверия и уровня доверия к процессу аудита информационной безопасности

Для проведения оценки доверия предлагается использование функции желательности Харрингтона, которая позволяет проводить однозначное соответствие количественных и качественных показателей произвольного процесса. Функция желательности (с односторонним ограничением задаётся уравнением,[23,24] [20]:

$$d = e^{-e^{-y'}} \quad (16)$$

где d – значение желательности в промежутке (0, 1), y' – значение частного признака, приведенное к промежутку [0, 7].

Ось координат Y называется шкалой частных показателей. Ось d – шкалой желательности. Промежуток эффективных значений на шкале частных показателей – [2, +5]. Для сдвига промежутка частных показателей в значения [0, 7] предлагается воспользоваться формулой:

$$d = e^{-e^{-(y'-2)}} \quad (17)$$

Шкала желательности содержит в себе ряд числовых промежутков, которым соответствует какой-либо лингвистический показатель желательности, который

23 Юсупова Г. Ф. Использование функции желательности в оценке уровня техносферной безопасности территории // Социально-экономические и технические системы: исследование, проектирование, оптимизация. – 2017. – №3 (76). – С. 67–81.
24 Пичкалев А. В. Обобщенная функция желательности Харрингтона для сравнительного анализа технических средств // Космические аппараты и технологии. – 2012. – №1 (1). – С. 25–28.

является качественной оценкой количественных значений желательности и несет в себе смысловую нагрузку, касающуюся значения желательности. В нашем случае лингвистические значения желательности будут обозначать степень зрелости процесса аудита информационной безопасности в качественной величине, а числовые значения желательности – степень зрелости процесса аудита в количественной величине.

Также функцию Харрингтона можно использовать при вычислениях значений желательности в несколько этапов. В таком случае финальное значение желательности называется обобщенной функцией желательности D, а промежуточные – частными функциями желательности d_i. Обобщенная функция желательности определяется как среднее арифметическое частных функций желательности:

$$D = \sqrt[n]{(d_1 * d_2 * ... * d_n)} \quad (18)$$

где n – число используемых показателей параметров сравнения для данной системы, причем число этих показателей может быть разным для разных систем.

За основу лингвистических показателей и промежутков показателей желательности (таблица 6) была взята процедура оценки соответствия по ГОСТ Р 57580.2–2018 «Безопасность финансовых (банковских) операций. Защита информации финансовых организаций. Базовый состав организационных и технических мер. Методика оценки соответствия», как самая проработанная процедура оценки соответствия, содержащая методику оценки и расчета степени соответствия мер защиты информации требования по защите информации.

Далее перейдем к расчету критериев доверия. Расчет критериев доверия основан на расчете частной функции желательности по формуле (17), где в качестве частных признаков желательности выступают соответствующие критерию метрики доверия M_{nm}.

2.3.1. Критерии доверия как частные функции желательности

Описанные в пп. 2.2.1–2.2.4 метрики доверия сгруппированы по 4 критериям доверия для многоступенчатой оценки процесса аудита информационной безопасности (рисунок 1). В данном пункте рассмотрим расчет числовых значений критериев доверия.

Исходя из формулы (16) для расчета значения каждого критерия необходимо определить вычисление приведенного значения y':

$$y_i' = k_i * \sum_{j=1}^{m} M_{ij}, \quad (19)$$

где m – количество измеряемых величин (метрики), j – порядковый номер величины, M_{ij} – j-ая измеряемая величина (метрика) i-ого критерия, k_i – корректирующий коэффициент, для приведенного значения в промежуток [0, 7].

$$k_i = \frac{y'_{max}}{y_{i\,max}}, \quad (20)$$

где y'_{max} – максимальное значение эффективного промежутка значений частного признака, $y_{i\,max}$ – максимальное значение суммы метрик i-ого критерия.

Получив из 20 значений метрик доверия 4 значения критерия доверия, можно перейти к финальному этапу расчета доверия к процессу аудита информационной безопасности. Уровень доверия к процессу аудита информационной безопасности основан на расчете обобщенной функции желательности по формуле (23).

2.3.2. Уровень доверия как обобщенная функция желательности

Общий уровень доверия к процессу аудита информационной безопасности рассчитывается из значений критериев доверия, рассчитываемых по пп. 2.3.1.

Исходя из формулы (18) приведения частных функций желательности к обобщенной сформирована

Таблица 6

Лингвистические значения оценки доверия

Обобщенный уровень доверия к аудиту, D	Уровень соответствия	Интерпретация
$D = 0$	Нулевой	Процедура аудита не выполняется
$0 < D \le 0{,}5$	Базовый	Процедура аудита выполняется на нерегулярной основе
$0{,}5 < D \le 0{,}7$	Базовый повышенный	Процедура аудита выполняется на регулярной основе и результат выполнения процесса задокументирован
$0{,}7 < D \le 0{,}85$	Средний	Процедура аудита выполняется, планируется, управляется и контролируется
$0{,}85 < D \le 1$	Высокий	Процедура аудита выполняется, планируется, управляется, измеряется при помощи количественных показателей (метрик) и постоянно совершенствуется

формула (21) для расчета уровня доверия к процессу аудита информационной безопасности исходя из значений критериев доверия:

$$D = \sqrt[4]{(d_1 * d_2 * d_3 * d_4)}, \quad (21)$$

где D – обобщенная функция желательности (уровень доверия к процессу аудита информационной безопасности), d_1 – частная функция желательности оценки доверия к полноте аудита, d_2 – частная функция желательности оценки доверия к качеству аудита, d_3 – частная функция желательности оценки доверия к своевременности аудита, d_4 – частная функция желательности оценки доверия к поставщику услуг аудита.

Итого, общий алгоритм оценки доверия к процессу аудита информационной безопасности выглядит следующим образом (рис. 3).

Смысл числового значения D базируется на таблице 6 – уровень контроля за состоянием системы защиты информации, означающий степень зрелости процессов аудита информационной безопасности. Уровень доверия к процессу аудита информационного обмена является одним из факторов доверия[25] к субъекту информационного обмена. Смысл показателя доверия к субъекту информационного обмена также базируется на таблице 6 – уровень злоумышленника, которому субъект информационного обмена способен противостоять в процессе информационного обмена с контрагентами.

25 ГОСТ 28195–89 Оценка качества программных средств. Общие положения.

Заключение

В ходе исследования был сформирован алгоритм проведения оценки доверия к процессу аудита информационной безопасности. Определены входные данные для процесса оценки доверия к процессу аудита информационной безопасности – 13 доказательств доверия из нормативно-правового обеспечения субъекта информационного обмена. Для расчета уровня доверия к процессу аудита информационной безопасности была использована функция желательности Харрингтона, а лингвистические показатели желательности сформированы на основе ГОСТ Р 57580.2–2018. ГОСТ Р 57580.2–201 взят за основу как самая проработанная методика проведения процедуры оценки соответствия. Определены 20 метрик и 4 критерия для оценки доверия как результаты оценки доказательств доверия. Описан метод расчета доверия к процессу аудита информационной

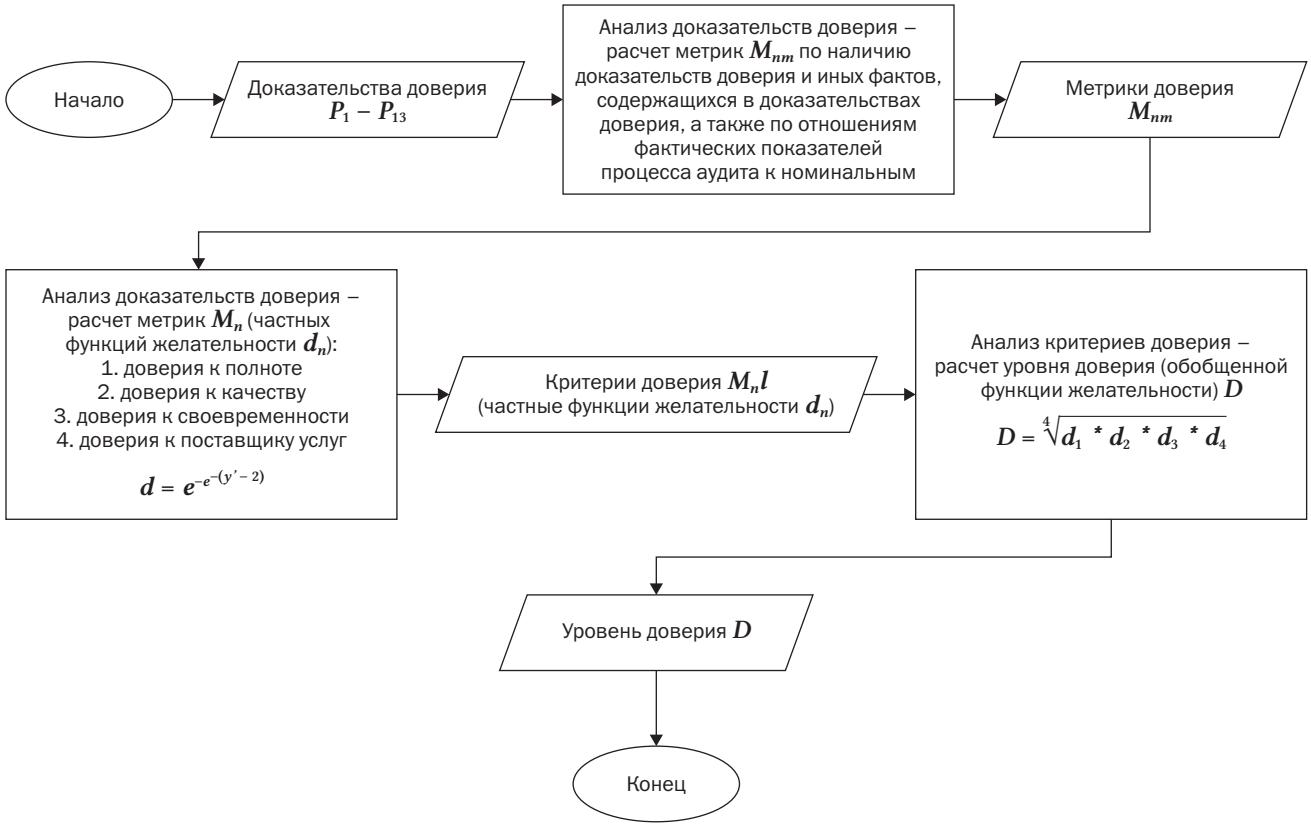


Рис. 3. Алгоритм оценки доверия к процессу аудита

безопасности на основе значений метрик и критериев доверия.

Сформированный алгоритм проведения оценки доверия к процессу аудита информационной безопасности является одной из составляющих процесса оценки доверия к субъектам информационного обмена и содержит объективные доказательства оценки уровня контроля за системой защиты информации информационных систем. Добросовестный подход к реализации процессов системы защиты информации и процессов информационной безопасности является одним из залогов способности субъекта информационного обмена противостоять злоумышленникам.

Алгоритм оценки доверия к процессу аудита информационной безопасности является одним из составляющих процесса оценки доверия к субъекту информационного обмена, охватывающего основные процессы информационной безопасности, включая процессы управления рисками, угрозами, уязвимостями, процессы аудита и менеджмента информационной безопасности. Далее в исследованиях планируется формирование имитационных моделей изучаемых процессов для оценки эффективности этих процессов, разработка методик оценки уровня доверия в различных условиях, разработка платформы доверенного взаимодействия, включающую модули оценки доверия.

Данная работа выполнена при финансовой поддержке Фонда поддержки проектов Национальной технологической инициативы (НТИ) в рамках реализации Программы Центра компетенций НТИ «Технологии доверенного взаимодействия» (договор от «14» декабря 2021 г. № 70-2021-00246).

Литература

1. *Кузнецова Н. М. Решение задачи автоматизации процессов защиты стратегически важных ресурсов предприятия от комплексных кибератак на основе анализа тактик злоумышленников / Н. М. Кузнецова, Т. В. Карлова, А. В. Бекмешов // Вестник Брянского государственного технического университета. 2020. №7 (92). URL: https://cyberleninka.ru/article/n/reshenie-zadachi-avtomatizatsii-protsessov-zaschity-strategicheski-vazhnyh-resursov-predpriyatiya-ot-kompleksnyh-kiber-atak-na-osnove (дата обращения: 12.02.2023).*
2. *Макаренко С. И. Тестирование на проникновение на основе стандарта NIST SP 800–115 // Вопросы кибербезопасности. – 2022. – №3 (49). – С. 44–57. DOI:10.21681/2311-3456-2022-3-44-49*
3. *К вопросу анализа нормативно-правовых документов по информационной безопасности автоматизированных систем органов внутренних дел Российской Федерации для оценки уровня их защищенности / Е. А. Рогозин, И. Г. Дровникова, А. О. Ефимов, В. Р. Романова // Вестник Дагестанского государственного технического университета. Технические науки. – 2022. – № 4 (49). – С. 97–103.*
4. *Селифанов В. В. Вопросы оценки доверия к системе управления рисками / В. В. Селифанов, В. В. Аникеева, И. А. Огнев // Безопасность цифровых технологий. – 2023. – № 1 (108). – С. 69–82. – DOI: 10.17212/2782–2230–2023–1–69–82.*
5. *Построение адаптивной трехуровневой модели процессов управления системой защиты информации объектов критической информационной инфраструктуры / А. С. Голдобина, Ю. А. Исаева, В. В. Селифанов, А. М. Климова, П. С. Зенкин // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2018. – №21. – С. 51–58.*
6. *Roy S. S. Enhanced trust management for building trustworthy social internet of things network / S.S. Roy, B.J.R. Sahu, S. Dash // IET Networks. – 2024. – № . – С. 1–11.*
7. *Access Control, Key Management, and Trust for Emerging Wireless Body Area Networks / A. S. Shahraki, H. Lauer, M. Grobler, A. Sakzad, C. Rudolph // Sensors. – 2023. – № 23 (24). – С. 1–32.*
8. *Брызгалов А. А. Применение концепции «нулевого доверия» для защиты коммерческой тайны на предприятии в условиях цифровизации / А. А. Брызгалов, П. А. Козырев, В. В. Ульянов // Вызовы цифровой экономики: технологический суверенитет и экономическая безопасность. – Брянск: ФГБОУ ВО «Брянский государственный инженерно-технологический университет» Инженерно–экономический институт, 2023. – С. 70–77.*
9. *Букирева Ю. М. Стратегия доступа к корпоративным сетям с применением модели нулевого доверия // Инновационные технологии: теория, инструменты, практика. – 2021. – №1. – С. 136–141.*
10. *Security of Zero Trust Networks in Cloud Computing: A Comparative Review / S. Sarkar, G. Choudhary, Sh. K. Shandilya, A. Hussain, H. Kim // Sustainability. – 2022. – №14. – С. 1–22.*
11. *Atencia M. Trust in networks of ontologies and alignments / M. Atencia, M. Al-Bakri, M.-C. Rousset // Knowledge and Information Systems. — 2013. — № 2 (42). — С. 1–27.*
12. *W. Al-shadood Enhancement the Security by Creating Ontology-Based Trust Management Using Semantic Web Tools // AlKadhum Journal of Science. — 2023. — № 2 (1). — С. 11–16.*
13. *Implementation of a Multi-Approach Fake News Detector and of a Trust Management Model for News Sources / C. Marche, I. Cabiddu, C. G. Castangia, L. Serreli // IEEE Transactions on Services Computing. – 2023. – № 6 (16). – С. 1–14.*
14. *Ан В. Р. Разработка алгоритма проведения аудита кибербезопасности / В. Р. Ан, В. А. Табакаева // МНСК-2021. Информационные технологии: материалы 59-й Международной научной студенческой конференции, Новосибирск, 12–23 апреля 2021 г. – Новосибирск, 2021. – С. 5. – EDN CAYHXE.*
15. *Макаренко С. И. Критерии и показатели оценки качества тестирования на проникновение // Вопросы кибербезопасности. – 2021. – №3 (43). – С. 43–57. DOI:10.681/2311-3456-2021-3-43-57*
16. *Ситская А. В. Вопросы аудита информационной безопасности / А. В. Ситская, В. В. Селифанов, П. А. Звягинцева // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 67–82.*
17. *Захахатнов В. Г. Функция желательности Харрингтона как критерий оптимального выбора зерносушилки / В. Г. Захахатнов, В. М. Попов, В. А. Афонькина // Известия Оренбургского государственного аграрного университета. – 2022. №2 (94). С. 110–114.*

ДЕНЕЖНЫЕ КРИТЕРИИ РИСКА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ НА ОСНОВЕ ПОДХОДА ОЦЕНКИ АКТИВОВ

Козырь Н. С.[1], Макарян А. С.[2], Оганесян Л. Л.[3]

DOI: 10.21681/2311-3456-2024-3-51-60

Цель исследования: *разработка критериев принятия риска информационной безопасности для подхода, основанного на оценке активов (ISO/IEC 27005).*

Методы исследования: *сделан анализ документов с участием ФСТЭК России, исследованы критерии принятия риска информационной безопасности (ИБ), определенных в стандарте ISO/IEC 27005 с учетом требований ГОСТ Р ИСО/МЭК 27001. На основе Международного стандарта аудита 320 даны рекомендации расчета уровня существенности ИБ, что должно стать основой для разработки критериев риска ИБ.*

Полученные результаты: *Критерии принятия риска для всех хозяйствующих субъектов должны базироваться на принципе существенности, которая составляет: 1% от совокупных активов; 1% от выручки или суммарных расходов (бюджет на год); 5% от прибыли (для коммерческих организаций). Показатель существенности может быть рассчитан для любой организации, включая бюджетные организации, где имеется показатель стоимости активов или сводный бюджет на год. Полученные выводы позволяют получить оценочную шкалу принятия рисков ИБ в денежном эквиваленте.*

Научная новизна: *исследование предлагает интеграцию экономических аспектов в процесс оценки критериев риска информационной безопасности, что позволяет организациям принимать обоснованные решения о приемлемости рисков, обосновывать бюджет ИБ, разрабатывать технико-экономическое обоснование проектов ИБ. Денежные критерии риска ИБ позволят реализовать подход ISO/IEC 27005 на основе активов.*

Вклад: *Козырь Н. С. – общая концепция исследования, структурирование, описание результатов, выводы; Макарян А. С. – систематизация нормативно-правовой документации в области рисков ИБ (ISO/IEC 27005, Методические документы и Приказы ФСТЭК); Оганесян Л. Л. – экономические аспекты риска ИБ (ГОСТ Р ИСО/МЭК 27001, МСА 320).*

Ключевые слова: *критерии риска ИБ, уровень существенности ИБ, экономика защиты информации, экономика риска ИБ, система менеджмента информационной безопасности, риск-менеджмент информационной безопасности, информационная безопасность, оценка риска ИБ, риски информационной безопасности.*

MONETARY INFORMATION SECURITY RISK CRITERIA BASED ON THE ASSET VALUATION APPROACH

Kozyr N. S.[4], Makaryan A. S.[5], Oganesyan L. L.[6]

The purpose: *to develop criteria for information security risk acceptance for an asset-based approach (ISO/IEC 27005).*

Research methods: *an analysis of documents with the participation of the FSTEC of Russia was made, the criteria for information security risk acceptance (IS) defined in the ISO/IEC 27005 standard were*

1 Козырь Наталья Сергеевна, кандидат экономических наук, доцент кафедры кибербезопасности и защиты информации, ФГБОУ ВО «Кубанский государственный технологический университет» (КубГТУ), г. Краснодар, Россия. E-mail: n_k_@mail.ru, ORCID 0000-0002-8323-0957.

2 Макарян Александр Самвелович, кандидат технических наук, доцент, заведующий кафедрой кибербезопасности и защиты информации, ФГБОУ ВО «Кубанский государственный технологический университет» (КубГТУ), г. Краснодар, Россия. E-mail: msanya@yandex.ru, ORCID 0000-0002-1801-6137.

3 Оганесян Левон Левонович, кандидат экономических наук, доцент, доцент кафедры кибербезопасности и защиты информации, ФГБОУ ВО «Кубанский государственный технологический университет» (КубГТУ), г. Краснодар, Россия. E-mail: oganesyan_levon@mail.ru, ORCID 0009-0004-5170-4515.

4 Natalia S. Kozyr, Ph.D. in Economics, Associate Professor of the Department of Cybersecurity and Information Protection, Kuban State Technological University (KubSTU), Krasnodar, Russia. E-mail: n_k_@mail.ru

5 Alexander S. Makaryan, Ph.D. in Engineering sciences, Associate Professor, Head of the Department of Cybersecurity and Information Protection, Kuban State Technological University (KubSTU), Krasnodar, Russia. E mail: msanya@yandex.ru

6 Levon L. Oganesyan, Ph.D. in Economics, Associate Professor of the Department of Cybersecurity and Information Protection, Kuban State Technological University (KubSTU), Krasnodar, Russia. E-mail: oganesyan_levon@mail.ru

studied, taking into account the requirements of GOST R ISO/IEC 27001. Based on the International Auditing Standard 320, recommendations are given for calculating the level of materiality of information security, which should become the basis for the development of information security risk criteria.

The results: *The criteria for risk acceptance for all business entities should be based on the principle of materiality, which is: 1% of total assets; 1% of revenue or total expenses (budget for the year); 5% of profit (for commercial organizations). The materiality indicator can be calculated for any organization, including budget organizations, where there is an asset value indicator or a consolidated budget for the year. The obtained conclusions allow us to obtain an estimated scale of information security risk acceptance in monetary terms.*

The novelty of the research: *the study suggests the integration of economic aspects into the process of assessing information security risk criteria, which allows organizations to make informed decisions about the acceptability of risks, justify the budget of information security, and develop a feasibility study of information security projects. The monetary risk criteria of the IB will allow the implementation of the ISO/IEC 27005 asset-based approach.*

Contribution: *Kozyr N. S. – the general concept of the study, structuring, description of the results, conclusions. Makaryan A. S. – systematization of regulatory and legal documentation in the field of information security risks (ISO/IEC 27005, Methodological documents and Orders of the Federal State Technical Committee); Oganesyan L. L. – economic aspects of information security risk (GOST R ISO/IEC 27001, ISA 320).*

Keywords: *information security risk criteria, information security materiality level, information security economics, information security risk economics, information security management system, information security risk management, information security, information security risk assessment, information security risks.*

Введение

Исследование восполняет пробел в части методического обоснования к разработке критериев риска на основе подхода оценки активов, что является актуальной задачей для обеспечения информационной безопасности (ИБ) организаций. В России обеспечение ИБ сфокусировано на выполнении требований законодательства и соответствующих приказов регуляторов, в основном это Федеральная служба по техническому и экспортному контролю (ФСТЭК), и исполнение всех приказов сводится к модели угроз с выбором соответствующих средств защиты информации. Вместе с этим, в стандарте ГОСТ Р 59503-2021 «Информационные технологии (ИТ). Методы и средства обеспечения безопасности. Менеджмент информационной безопасности. Экономика информационной безопасности организации» обозначен подход по внедрению экономической модели как ключевого элемента процесса принятия решений в соответствии с комплексом стандартов ИСО/МЭК 27000 (ГОСТ Р 59503-2021 идентичен ISO/IEC TR 27016).

В России сохраняется проблема в части определения «границ» безопасности информационной системы на уровне национальных стандартов ГОСТ Р и регламента в сфере информационной безопасности[7], а значимость экономических аспектов информационной безопасности недооценивается специалистами в сфере защиты информации [1]. Напротив, современные тенденции в области системы менеджмента информационной безопасности характеризуются обновлением существующих, и разработкой новых стандартов ИСО/МЭК 27xxx, охватывающих актуальные сферы технологического развития глобального мира с акцентом на денежной стоимости риска и оценке активов.

Так, в зарубежных публикациях приводятся данные об обеспечении ИБ на основе комплексного управления системой риск-менеджмента ISO/IEC 27xxx [2], что в целом позволяет повысить уровень организационного развития хозяйствующего субъекта [3]. В трудах российских авторов сделано обзор зарубежных методик риск-менеджмента [4] и возможность их применения в российской практике обеспечения ИБ [5]. При этом отсутствие возможности денежной оценки рисков – распространенная проблема, в том числе, для зарубежных методик риск-менеджмента ИБ (COBIT for Risk, CRAMM, FRAP, OCTAVE), в работах ученых [6, 7].

В России анализ рисков рассматривается в контексте разработки модели угроз [8], с соответствующим анализом программного обеспечения [9]. В работе ученых (С. А. Никулин, С. С. Никулин) приведен математический аппарат определения риска, который основывается на принципах вероятности угроз и ущерба, без критериев содержательного наполнения (ГОСТ Р ИСО/МЭК 27005-2010), а негативные события обозначены буквенными параметрами

7 Максименко В. Н., Ясюк Е. В. Основные подходы к анализу и оценке рисков информационной безопасности // Экономика и качество систем связи. 2017. № 2(4). С. 42–48.

без количественного расчета[8]. Также есть научные публикации, посвященные решению прикладных задач комплексного управления рисками (Н. И. Касперская [10], М. М. Путято и А. С. Макарян [11]).

Говоря о прикладных решениях в части критериев оценки безопасности информационных технологий, также следует отразить проблему с методическим обеспечением, которое характеризуется эволюционным отставанием от зарубежных аналогов. Так, в национальных стандартах России имеется три части ГОСТ Р 15408 (Критерии оценки безопасности информационных технологий), при этом за рубежом применяется пять частей ISO/IEC 15408, которые прошли два этапа эволюции. Действующие стандарты ГОСТ Р 15408 для применения нуждаются в гармонизации с ГОСТ Р ИСО/МЭК 27005[9]. В настоящее время нет универсальной методики, которая решила бы задачи мониторинга, анализа, оценки и предотвращения рисков и угроз системы защиты информации [12]. Основной пробел в методическом обеспечении оценки рисков ИБ является денежная оценка критериев, которая позволит сформировать систему риск-менеджмента с учетом ресурсных возможностей организации. В этой связи представленное исследование посвящено экономике критериев риска ИБ и определению существенности в денежном эквиваленте, что позволит реализовать подход оценки риска на основе активов.

Методология исследования

Методология исследования «Денежные критерии риска информационной безопасности на основе подхода оценки активов» предопределена следующей логикой: обеспечение ИБ для хозяйствующих субъектов преимущественно базируется на требованиях регулятора ФСТЭК, где исполнением приказов является составление модели угроз и выбор соответствующих средств защиты информации; ФСТЭК принимает активное участие в составе Технического комитета по стандартизации ТК 362 «Защита информации» в разработке национальных стандартов, и проект ГОСТ Р ИСО/МЭК 27005:2022 размещен на официальном сайте (в документе говорится о критериях риска, основанных на подходе оценки активов); для формирования методической основы определения критериев риска необходимо провести мониторинг действующих документов в этой области, включая ГОСТ Р ИСО/МЭК 27001:2021.

Таким образом, в работе рассмотрена методика оценки угроз безопасности информации ФСТЭК России на предмет содержания экономических аспектов риска, сделан анализ существующих критериев оценки информационной безопасности. Особое внимание уделено национальному стандарту ГОСТ Р ИСО/МЭК 27001-2021 в части раздела оценки рисков. Основой для рекомендаций послужило исследование денежного аспекта критериев принятия рисков в ИСО/МЭК 27005. В результате были сформированы пороговые значения критериев риска информационной безопасности в денежном эквиваленте на основе международного стандарта аудита 320.

Используемые сокращения в публикации:

СМИБ система менеджмента информационной безопасности;

МСА Международный стандарт аудита;

ОКУД Общероссийский классификатор управленческой документации.

Методика оценки угроз безопасности информации ФСТЭК России: экономические аспекты риска

В этом разделе представлен детальный анализ «Методики оценки угроз безопасности информации» ФСТЭК России на предмет содержания слова «риск» (в документе упоминание «риск» встречается

Раздел методики угроз ИБ	Содержание компонента риска
2.3. Исходными данными для оценки угроз безопасности информации являются:	ж) результаты оценки рисков (ущерба), проведенной обладателем информации и (или) оператором
2.7 Для оценки угроз безопасности информации рекомендуется привлекать специалистов, обладающих следующими знаниями и умениями:	а) основ оценки рисков, а также основных рисков от нарушения функционирования систем и сетей и нарушения безопасности обрабатываемой информации (далее – информационные риски)
	ж) оценивать информационные риски
3.2. Исходными данными для определения негативных последствий от реализации угроз безопасности информации являются:	д) результаты оценки рисков (ущерба), проведенной обладателем информации или оператором
3.3. На основе анализа исходных данных определяются событие или группа событий, наступление которых в результате реализации (возникновения) угроз безопасности информации может привести к:	в) возникновению финансовых, производственных, репутационных или иных рисков (видов ущерба) для обладателя информации, оператора

Рис. 1. Содержание компонентов риска в методике угроз ИБ в разделе 2 (порядок оценки угроз) и разделе 3 (определение негативных последствий)

8 Никулин С. А., Никулин С. С. Методика количественной оценки величины риска обеспечения информационной безопасности автоматизированных систем управления и связи // Вестник Воронежского института ФСИН России. 2016. № 1. С. 44–51.

9 Барабанов А. В., Марков А. С., Цирлов В. Л. 28 магических мер разработки безопасного программного обеспечения // Вопросы кибербезопасности. 2015. № 5(13). С. 2–10.

34 раза)[10]. Следует отметить, что в ряде случаев вместе с риском встречается рекомендация по оценке финансового компонента возможных последствий от нежелательных инцидентов информационной безопасности. Все ссылки в этом разделе относятся к структуре анализируемого документа.

В методическом документе «Методика оценки угроз безопасности информации» ФСТЭК России говорится о необходимости оценки риска и его последствий (рис. 1).

Однако следует отметить и то, что в ситуации отсутствия результатов оценки рисков (ущерба) методический документ предлагает вариант определения возможных негативных последствий от реализации угроз ИБ на основе экспертной оценки специалистов (раздел 3.4). Вместе с этим, результаты оценки ущерба (рисков) относятся к категории исходных данных для определения возможных актуальных нарушителей (раздел 5.1.2). В описании актуальных нарушителей (раздел 5.1.4) говорится о том, что к таковым относятся все, чьи действия «могут привести к определенным для систем и сетей негативным последствиям и соответствующим рискам (видам ущерба)». На рис. 2 представлены компоненты риска, которые содержатся в приложении 1, 2 и 3 методического документа ФСТЭК России.

Примеры сопоставления возможных целей реализации угроз безопасности информации с видами ущерба (риска) и возможными негативными последствиями о реализации угроз представлены в приложении 5 Методики угроз ФСТЭК России. В Приложении 7 сделано соотнесение целей видам риска (ущерба) по видам нарушителей и возможных негативных последствий на объекты воздействия.

Несмотря на то, что риски – неотъемлемый компонент методического документа по оценке угроз, ФСТЭК России не дает конкретных инструкций по их оценке. Вместе с этим, о важности анализа риска говорится в руководящем документе по разработке профилей защиты и заданий по безопасности[11]: «если анализ рисков не выполнен должным образом, объект оценки будет не в состоянии обеспечить

10 Методика оценки угроз безопасности информации (утв. ФСТЭК России 05.02.2021) [электронный ресурс]. Режим доступа: https://fstec.ru/files/495/---5--2021-/891/---5--2021-.pdf. (дата обращения: 01.02.2024)

11 Руководство по разработке профилей защиты и заданий по безопасности. Руководящий документ (Гостехкомиссия России, 2003 год). [электронный ресурс]. Режим доступа: https://fstec.ru/files/576/--2003-482/1040/--2003-.pdf. (дата обращения: 01.02.2024).

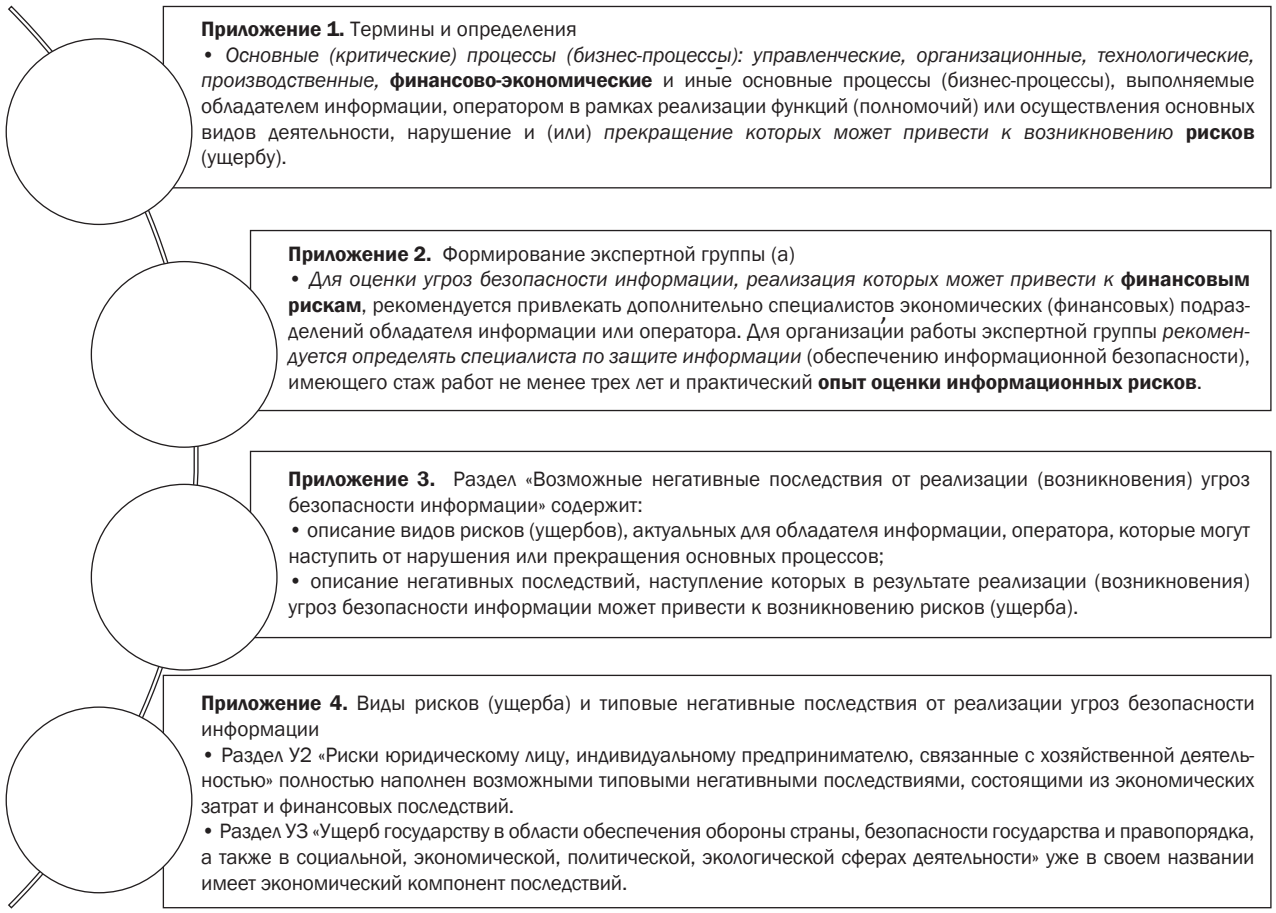


Рис. 2. Обзор приложений методики угроз ФСТЭК России на предмет описания категории риска

адекватную защиту, в результате чего активы организации могут остаться подверженными соответствующему риску». Наряду с этим отмечается, что рекомендации по организации процесса идентификации угроз активам – это один из самых трудоемких этапов анализа риска организации, и не включены в руководство по разработке профилей защиты и заданий по безопасности, в связи с чем, в документе изложены только общие принципы идентификации угроз.

За рубежом аспекты риска информационной безопасности являются предметом национальной системы стандартизации (NIST – США, BS – Англия), где активно переведены (или адоптированы) международные стандарты ISO/IEC, включая серию 27xxx «Менеджмент риска информационной безопасности». Несмотря на развитую систему стандартизации РФ, тема рисков имеет локальный характер, что осложняет развитие методического обеспечения информационной безопасности для хозяйствующих субъектов.

Критерии оценки безопасности ИТ и менеджмент риска ИБ в национальных стандартах ГОСТ Р

Важно отметить, что ФСТЭК России – ключевая организация в регулировании ИБ с активным участием в разработке национальных стандартов РФ (в составе деятельности технического комитета по стандартизации «Защита информации», ТК 362)[12].

Полный перечень активности ФСТЭК России в составе ТК по разработке национальных стандартов ГОСТ Р представлен на официальном сайте (раздел «стандарты), на рис. 3 представлены документы, которые содержат критерии оценки информационной безопасности (ГОСТ Р 15408) и категорию «риск» (ГОСТ Р 27005). Несмотря на то, что стандарты ГОСТ Р 15408 не соотносятся напрямую с риском информационной безопасности, но в национальной системе стандартизации нет других, которые бы содержали «критерии» оценки безопасности, что подтверждает общую проблему для разработки критериев риска – отсутствие методической документации.

Иллюстрация показывает, что в России применяются национальные стандарты ГОСТ Р 15408, которые отстают по составу и эволюционному развитию от международных ISO/IEC 15408, что в свою очередь отражается на методическом обеспечении оценки рисков информационной безопасности. Эволюционное отставание национальной системы стандартизации было отмечено в работе, посвященной анализу новых пакетов нормативных методических документов ФСТЭК России[13].

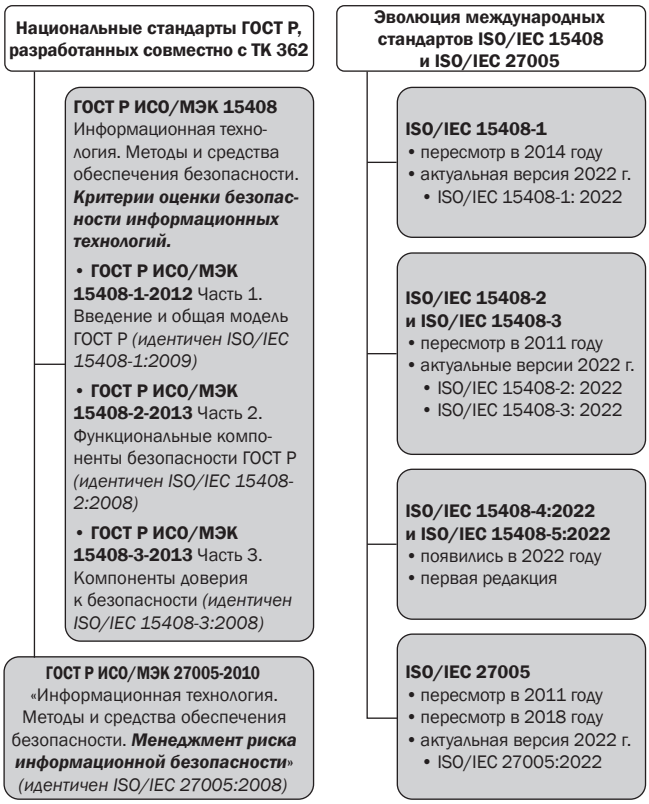


Рис. 3. Сравнительный анализ национальных стандартов ГОСТ Р и актуальных редакций международных стандартов ISO/IEC

Так, например, в международный стандарт ISO/IEC 15408-1:2009 в 2014 году были внесены правки, а в 2022 году принята новая версия стандарта[14].

Аналогичные эволюционные изменения коснулись второй и третьей части ISO/IEC 15408, в 2011 году были исправлены ошибки. В настоящее время действует международные стандарты 2022 года ISO/IEC 15408-2:2022[15] и ISO/IEC 15408-3:2022[16]. В 2022 году появилась четвертая и пятая части зарубежного стандарта ISO/IEC 15408, которые посвящены описанию методов и мероприятий оценки безопасности информационных технологий[17] и определению пакетов требований безопасности[18].

12 Марков А. С., Цирлов В. Л. Структурное содержание требований информационной безопасности // Мониторинг правоприменения. 2017. № 1(22). С. 53–61. DOI 10.21681/2412-8163-2017-1-53-61.

13 Барабанов А. В., Марков А. С., Цирлов В. Л. Оценка соответствия средств защиты информации «Общим критериям» // Информационные технологии. 2015. Т. 21. № 4. С. 264–270.

14 ISO/IEC 15408-1:2022 Information security, cybersecurity and privacy protection. Evaluation criteria for IT security. Part 1: Introduction and general model [электронный ресурс]. Режим доступа: https://www.iso.org/ru/standard/72891.html (дата обращения: 01.02.2024).

15 ISO/IEC 15408-2:2022 Information security, cybersecurity and privacy protection. Evaluation criteria for IT security. Part 2: Security functional components [электронный ресурс]. Режим доступа: https://www.iso.org/ru/standard/72892.htm (дата обращения: 01.02.2024).

16 ISO/IEC 15408-3:2022 Information security, cybersecurity and privacy protection. Evaluation criteria for IT security. Part 3: Security assurance components [электронный ресурс]. Режим доступа: https://www.iso.org/ru/standard/72906.html (дата обращения: 01.02.2024).

17 ISO/IEC 15408-4:2022 Information security, cybersecurity and privacy protection. Evaluation criteria for IT security. Part 4: Framework for the specification of evaluation methods and activities [электронный ресурс]. Режим доступа: https://www.iso.org/standard/72913.html (дата обращения: 01.02.2024).

18 ISO/IEC 15408-5:2022 Information security, cybersecurity and privacy protection. Evaluation criteria for IT security. Part 5: Pre-defined packages of security requirements [электронный ресурс]. Режим доступа: https://www.iso.org/ru/standard/72917.html (дата обращения: 01.02.2024).

Особого внимания заслуживает национальный стандарт ГОСТ Р ИСО/МЭК 27005-2010 (на основе ISO/IEC 27005:2008), который нуждается в пересмотре. Международный стандарт ISO/IEC 27005 за 15 лет трижды изменился, в 2011 и 2018 года пересмотрен, а в 2022 году принята новая редакция. Этот документ применим ко всем организациям, независимо от типа, размера или сектора. Действующая редакция ISO/IEC 27005:2022 направлена не только на выполнение требования стандарта ISO/IEC 27001 (действия по устранению рисков информационной безопасности), но и позволяет выполнять мероприятия по управлению рисками информационной безопасности (в частности, оценку рисков информационной безопасности и их обработку)[19].

Надо отметить, что осенью 2023 года на сайте ФСТЭК появился проект национального стандарта ГОСТ Р ИСО/МЭК 27005-2022 (идентичен ISO/IEC 27005:2022)[20]. В настоящее время не обозначены сроки рассмотрения и перспективы принятия проекта, тем не менее, далее в исследовании рассмотрен этот документ, на предмет выявления критериев риска информационной безопасности на основе подхода оценки активов. Учитывая, что стандарты 27xxx серии раскрывают те или иные аспекты ГОСТ Р ИСО/МЭК 27001, контекст риска необходимо определить место и роль риск-менеджмента в стандарте, на основе которого осуществляется сертификация в соответствии с ISO/IEC 27001:2022.

Национальный стандарт ГОСТ Р ИСО/МЭК 27001-2021: оценка рисков информационной безопасности

В настоящее время в РФ действующей редакцией является ГОСТ Р ИСО/МЭК 27001-2021 «Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Требования» (национальный

19 ISO/IEC 27005:2022 Information security, cybersecurity and privacy protection. Guidance on managing information security risks [электронный ресурс]. Режим доступа: https://www.iso.org/standard/80585.html (дата обращения: 01.02.2024).

20 Проект ГОСТ Р ИСО/МЭК 27005-2022 Информационная безопасность, кибербезопасность и защита частной жизни. Руководство по управлению рисками информационной безопасности. Требования и руководства [электронный ресурс]. Режим доступа: https://fstec.ru/files/1135/-------27005/2138/----27005----------.pdf (дата обращения: 01.02.2024).

Оценка рисков информационной безопасности (ГОСТ Р ИСО/МЭК 27001-2021, раздел 6.1.2.)		
Организация должна определить и внедрить процесс оценки рисков информационной безопасности, который позволяет:	6.1.2 a) ***устанавливать и поддерживать критерии рисков информационной безопасности,*** включая	*1)* ***критерии принятия рисков информационной безопасности;***
		2) ***критерии для проведения оценки рисков информационной безопасности;***
	6.1.2 b) обеспечивать уверенность в том, что повторные оценки рисков информационной безопасности дают непротиворечивые, достоверные и сопоставимые результаты;	
	6.1.2 c) идентифицировать риски информационной безопасности, т.е.:	1) применять процесс оценки рисков информационной безопасности для идентификации рисков, связанных с нарушением конфиденциальности, целостности и доступности информации в рамках области действия системы менеджмента информационной безопасности;
		2) идентифицировать владельцев рисков информационной безопасности;
	6.1.2 d) проводить анализ рисков информационной безопасности, т.е.:	1) оценивать потенциальные последствия, которые могут произойти в результате реализации рисков информационной безопасности, идентифицированных в соответствии с 6.1.2 c) 1);
		2) оценивать реальную вероятность реализации рисков информационной безопасности, идентифицированных в соответствии с 6.1.2 c) 1);
		3) определять уровни рисков информационной безопасности;
	6.1.2 e) ***оценивать риски информационной безопасности,*** т.е.:	*1)* ***сравнивать результаты анализа рисков информационной безопасности с критериями рисков,*** *установленными в соответствии с 6.1.2 a);*
		2) ***определять приоритетность обработки проанализированных рисков информационной безопасности.***

Рис. 4. Содержание раздела 6.1.2 «Оценка рисков информационной безопасности» ГОСТ Р ИСО/МЭК 27001-2021

стандарт идентичен ISO/IEC 27001:2013). Несмотря на то, что в 2022 году вышла новая версия международного стандарта ISO/IEC 27001:2022, раздел «Оценка рисков информационной безопасности» (6.1.2) не содержит существенных изменений в сравнении с 2013 годом, в этой связи на иллюстрации показаны данные ГОСТ Р ИСО/МЭК 27001-2021 (рис. 4).

Так, структура оценки рисков ИБ начинается с необходимости установления критериев, которые позволяют оценить риски и в дальнейшем – использовать для принятия (раскрытие раздела анализа рисков содержится в ИСО/МЭК 27005). Завершается раздел 6.1.2 тоже процедурой анализа рисков на основе установленных критериев. Таким образом, слабым звеном в риск-менеджменте являются критерии, которые организации должны установить самостоятельно, и использовать в своей деятельности.

Денежный аспект критериев принятия рисков информационной безопасности в ИСО/МЭК 27005

Для анализа критериев принятия риска в представленном исследовании рассмотрен проект национального стандарта ГОСТ Р ИСО/МЭК 27005-2022, т.к. содержит новый подход, основанный на активах. Свод денежных и финансовых критериев риска информационной безопасности представлен в табл. 1.

Наряду с этим, в критериях последствий (раздел 6.4.3.2) предлагается использование логарифмической шкалы денежных последствий с комбинированием оценки уровня последствий в других областях (без финансовых аспектов). Также в документе говорится о том, что критерии принятия риска ИБ должны быть установлены, в том числе, с учетом финансовых ограничений (раздел 6.4.2).

В разделе «Мониторинг и анализ факторов, влияющих на риски» (раздел 10.5.2) акцентируется внимание на том, что риски не являются статичными, в связи с чем требуется постоянный пересмотр всех оценочных факторов, которые в том числе включают в себя появление новых активов с корректировкой их стоимости.

В приложении А описаны критерии риска информационной безопасности, где финансовые потери в денежных единицах и условная частота возникновения рискового события – неотъемлемые компоненты принятого порогового значения, выше которого риски считаются неприемлемыми (А.1).

Подход на основе активов подразумевает обязательную привязку оценки рисков с первичными бизнес активами и вспомогательными активами. В этой связи важно определить взаимосвязи между активами и понять их ценность, т.к. неправильная оценка стоимости активов приведет к неправильной оценке последствий, связанных с риском (А 2.2). Таким образом, критерии риска должны быть связаны со стоимостью активов, а пороговые значения для принятия риска необходимо устанавливать в денежном эквиваленте.

Пороговые значения критериев риска информационной безопасности в денежном эквиваленте

Денежный эквивалент пороговых значений риска информационной безопасности необходимо установить в соответствии со стандартом, который применяется в практике аудита финансовой отчетности РФ – МСА 320 «Существенность при планировании и проведении аудита» (МСА 320)[21].

По своей сути, подходы для определения уровня существенности в аудите и пороговые денежные критерии риска ИБ имеют одинаковую экономическую природу. Для определения критериев ИБ необходимо за основу взять один из трех показателей, которые присущи всем хозяйствующим субъектам независимо от формы собственности и вида экономической деятельности:

- 1% от совокупных активов организации;
- 1% от выручки или 1% от годового бюджета плановых расходов (для государственных и некоммерческих организаций);
- 3% от прибыли до налогообложения (или 5% от чистой прибыли).

Выбранный показатель является основой для расчета уровня существенности ИБ, который имеет денежный эквивалент для любого хозяйствующего субъекта. В табл. 2 представлены рекомендации по выбору целевого показателя для расчета критериев риска ИБ. Рекомендуется выбирать тот показатель, который даст максимальное значение.

Уровень существенности ИБ – это предельная величина совокупных рисков в денежной стоимости. Для формирования критериев риска в организации должен быть принят локальный нормативный акт с определением уровня значимости риска в привязке к денежной стоимости уровня существенности ИБ. Организации должны опираться на показатель «уровень существенности ИБ» при принятии решений о том, какие риски информационной безопасности им следует принять и какие меры по управлению рисками следует реализовать.

Использование конкретных значений критериев существенности позволяет организациям более четко определить и оценить риски информационной безопасности, что способствует более эффективному управлению ими и снижению потенциальных убытков.

21 Международный стандарт аудита 320 «Существенность при планировании и проведении аудита» [электронный ресурс]. Режим доступа: https://minfin.gov.ru/ru/document/?id_4=116584 (дата обращения 01.02.2024).

Таблица 1

Денежно-финансовые содержательные компоненты в документе «Проект национального стандарта ГОСТ Р ИСО/МЭК 27005-2022»

п/п	Раздел и название 27005-2022	Связь с 27001:2022	Денежный и финансовый компоненты содержания рисков ИБ в проекте стандарта ГОСТ Р ИСО/МЭК 27005:2022
1	6.4 Установление и поддержание критериев риска информационной безопасности 6.4.3 Критерии для оценки риска информационной безопасности		
1.1	6.4.3.2 Критерии последствий	6.1.2 a) 2)	При определении критериев последствий следует особенно учитывать возможность (опасность): f) потери деловой и ***финансовой ценности***.
			Максимальная сумма, которую организация готова списать в течение финансового года, и ***минимальная сумма*** за тот же период, которая вынудила бы ее к ликвидации, могут создать реалистичные верхние и нижние пределы шкалы критериев последствий организации, которые представлены в ***денежном выражении***.
1.2	6.4.3.4 Критерии для определения уровня риска	6.1.2 a) 2)	Критерии уровня риска могут быть качественными (например, очень высокий, высокий, средний, низкий) или количественными (например, выраженными в терминах ожидаемой ***величины денежных потерь***, гибели людей или доли рынка за определенный период времени).
			Риски могут быть количественно определены как ***ожидаемый годовой убыток***, т. е. ***средняя денежная стоимость последствий*** за год, принятых в течение следующего года.
2	7.3 Анализ рисков информационной безопасности		
2.1	7.3.1 Общие положения	6.1.2 d) 1) 6.1.2 d) 2)	Методы анализа рисков, учитывающие последствия и их вероятность, могут быть обоснованы значениями показателей: a) качественными, которые используют качественную шкалу квалификационных признаков (например, высокий, средний, низкий); b) количественными, которые используют количественную шкалу с числовыми значениями (например, ***денежная стоимость***, частота или вероятность возникновения); c) полуколичественными, которые используют совокупность качественных и количественных шкал с присвоенными значениями.
2.2	7.3.4 Определение уровней риска	6.1.2 d) 3)	Уровень риска определяется как комбинация оцененной вероятности и оцененных последствий для соответствующих сценариев риска. Альтернативные расчеты могут включать ***стоимость актива***, а также их вероятность и оценку последствий.
2.3	7.4.1 Сравнение результатов анализа рисков с критериями риска	6.1.2 e) 1)	Уровни риска могут быть согласованы на основе консенсуса между владельцами рисков, деловыми и техническими специалистами. Важно, чтобы владельцы рисков хорошо понимали ***последствия материализации рисков***, за которые они несут персональную ответственность
3	9 Реализация		
3.1	9.1 Процесс оценки риска информационной безопасности	6.1.2 a)	Если существует годовой ***бюджетный цикл***, то могут потребоваться ***запросы на финансирование*** в определенные периоды бюджетного года.
			Следует заранее запланировать оценку риска: a) своевременно подавать предложения по обработке рисков и заявки на ***финансирование***; b) заранее провести переоценку рисков в соответствии с предполагаемыми ***бюджетными ассигнованиями***.
4	10 Применение взаимозависимых процессов СМИБ		
4.1	10.7 Корректирующие действия	н/д	План обработки рисков должен быть пересмотрен с учетом: выявленных трудностей при внедрении средств контроля (например, технические или ***финансовые проблемы***, несоответствия внутренним или внешним факторам, таким как соображения конфиденциальности).
4.2	10.8 Постоянное совершенствование	н/д	Организация должна регулярно проверять критерии, используемые для измерения риска. Деятельность по мониторингу и обзору должна охватывать (но не ограничиваться): правовой и экологической сферой; сферой конкуренции; подход к оценке риска; ***стоимость активов*** и их категории; критерии последствий; критерии вероятности; критерии оценивания риска; критерии принятия риска; общую ***стоимость владения***; необходимые ресурсы.

Таблица 2

Показатели для расчета уровня существенности риска информационной безопасности

п/п	Показатель	Применимость для хозяйствующих субъектов	Пояснение по расчету критерия
1	1% от совокупных активов	Крупный корпоративный бизнес с широко развитой сетью структурных подразделений	*Для коммерческих организаций:* Документ «Бухгалтерский баланс (ОКУД 0710001)», рассчитать 1% от показателя «БАЛАНС» (код строки 1600 или 1700, т.к. значения равны)
2	1% от выручки или 1% от годового бюджета плана расходов	Некоммерческие организации, Малый и средний бизнес, Индивидуальные предприниматели, Крупный корпоративный бизнес с высокой нормой прибыли (низкая доля расходов, например – управляющая компания). Ниже приведены примеры хозяйствующие субъекты	У каждой организации есть отчет о финансовых результатах, или утвержденный бюджет доходов, или перечень утвержденных расходов. Расчетный показатель не обязательно называется «выручка», по сути – для расчета берется объем финансового потока поступивших денежных средств за календарный год. Рассчитать 1% от показателя:
		– Коммерческие организации	«Выручка» (код строки 2110),: документ «Отчет о финансовых результатах (ОКУД 0710002)»
		– Некоммерческие организации (НКО)	«Текущие расходы – всего» (код строки 41), документ «Сведения о деятельности некоммерческой организации (ОКУД 0608032)»
		– Государственные внебюджетные фонды	«Доходы бюджета – всего» (код строки 010), или «Расходы бюджета - всего» (код строки 200) документ «Отчеты об исполнении бюджетов государственных внебюджетных фондов (ОКУД 0503317)»
		– Индивидуальные предприниматели и малый бизнес без обязательных форм отчетности	Объем поступивших денежных средств на официальный расчетный счет за календарный год
		– Другие организации	Поступившие денежные средства за календарный год с учетом источников финансирования дефицита бюджетов
3	5% от чистой прибыли	Коммерческие организации с низкой долей основных средств и высокой нормой прибыли	*Для коммерческих организаций:* Документ «Отчет о финансовых результатах (ОКУД 0710002)», рассчитать 5% от показателя «Чистая прибыль» (код строки 2400)

Заключение

Практическая значимость исследования состоит в том, что оно предоставляет организациям конкретные рекомендации по определению критериев принятия риска, основанных на принципе существенности и оценке активов. Полученные значения критериев существенности (1% от совокупных активов, 1% от выручки или суммарных расходов на год, 5% от чистой прибыли) могут быть использованы в практике для технико-экономического обоснования проектов и обоснования бюджета на информационную безопасность. Уровень существенности ИБ – это расчетная величина, имеет конкретное значение для каждой организации и выражена в денежной форме.

Так, например, компоненты риска из методики оценки угроз безопасности информации ФСТЭК России должны быть просуммированы, и итоговое значение необходимо сравнить с уровнем существенности ИБ в конкретной организации. Если сумма стоимости рисков ниже, чем принятый уровень существенности, значит, сохранится общее финансовое благополучие организации, и возможные инциденты не окажут влияние на непрерывность деятельности хозяйствующем субъекта.

Уровень существенности ИБ применим для реализации национальных стандартов ГОСТ Р 15408, т.к. предоставляет обоснование для сравнения критериев оценки информационной безопасности с финансовыми возможностями организации. Для практической реализации национального стандарта ГОСТ Р ИСО/МЭК 27001-2021 – расчет уровня существенности позволяет определить критерии принятия и проведения оценки рисков информационной безопасности.

Таким образом, решается методическое обеспечение реализации подхода, основанного на активах, и принять в РФ национальный стандарт, который будет идентичен актуальному документу ISO/IEC 27005:2022 «Информационная безопасность, кибербезопасность и защита частной жизни. Руководство по управлению рисками информационной безопасности. Требования и руководства».

Литература

1. *Козырь Н. С., Оганесян Л. Л. Экономические аспекты информационной безопасности. – Москва: ЮРАЙТ, 2023. 131 с.*
2. *Razikin Kh., Soewito B. Cybersecurity decision support model to designing information technology security system based on risk analysis and cybersecurity framework // Egyptian Informatics Journal. 2022. Vol. 23. No 3. P. 383-404. DOI 10.1016/j.eij.2022.03.001.*
3. *Schmid M., Pape S. Aggregating corporate information security maturity levels of different assets // IFIP Advances in Information and Communication Technology. 2020. Vol. 576. P. 376-392. DOI: 10.1007/978-3-030-42504-3_24.*
4. *Маслова М А. Научно-методические рекомендации по регулированию рисков нарушения информационной безопасности // Информация и безопасность. 2022. Т. 25. № 4. С. 513–520. DOI 10.36622/VSTU.2022.25.4.005.*
5. *Волкова Л. В., Макарова Д. В., Докучаев В. А. Использование метода CRAMM для оценки информационных рисков // Телекоммуникации и информационные технологии. 2021. Т. 8. № 1. С. 103–109.*
6. *Кортнев К. Методики управления рисками информационной безопасности и их оценки (часть 1, 14.05.2018) [электронный ресурс]. Режим доступа: https://safe-surf.ru/specialists/article/5193/587932/ (дата обращения 16.10.2023).*
7. *Кортнев К. Методики управления рисками информационной безопасности и их оценки (часть 2, 22.05.2018) [электронный ресурс]. Режим доступа: https://safe-surf.ru/specialists/article/5194/587935/?sphrase_id=45664 (дата обращения 16.10.2023).*
8. *Повышев А. А., Соколов А. Н., Мищенко Е. Ю. Универсальная классификация угроз безопасности информации и её применение для разработки модели угроз и оценки рисков // Вестник УрФО. Безопасность в информационной сфере. 2023. № 3(49). С. 68-80. DOI 10.14529/secur230307.*
9. *Баранова Е. К., Мурзакова А. А., Мурзакова Е. А. Сравнительный анализ программного обеспечения для анализа рисков информационной безопасности в соответствии с ГОСТ Р ИСО/МЭК 27005-10 // Информационные технологии и вычислительные системы. 2019. № 2. С. 75–83. DOI 10.14357/20718632190208.*
10. *Касперская Н. И. Анализ больших данных в ИБ предприятий. Перспективы развития // Защита информации. Инсайд. 2019. № 3(87). С. 34–43.*
11. *Путято М. М., Макарян А. С. Подходы к построению адаптивной системы защиты на основе корреляционного анализа статистических характеристик инцидентов информационной безопасности // Электронный сетевой политематический журнал «Научные труды КубГТУ». 2022. № 2. С. 148–162.*
12. *Козырь Н. С. Методические подходы риск-менеджмента информационной безопасности // Электронный сетевой политематический журнал «Научные труды КубГТУ». 2023. № 4. С. 99–109.*

ИССЛЕДОВАНИЕ СОСТЯЗАТЕЛЬНЫХ АТАК НА РЕГРЕССИОННЫЕ МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ В БЕСПРОВОДНЫХ СЕТЯХ 5G

Легашев Л. В.[1], Жигалов А. Ю.[2]

DOI: 10.21681/2311-3456-2024-3-61-67

Цель исследования: *Исследование влияния состязательных атак на метрики качества регрессионных моделей машинного обучения.*

Метод исследования: *Эмуляция данных распространения сигнала в MIMO системах, синтез состязательных примеров, выполнение состязательных атак на модели машинного обучения, обучение бинарных классификаторов для обнаружения состязательных аномалий в данных.*

Результат исследования: *В статье проведена генерация сценария и исследовательский анализ набора данных с помощью эмулятора DeepMIMO. Выполнена состязательная атака с максимизацией знака градиента методом FGSM. Выполнено экспериментальное сравнение бинарных классификаторов для обнаружения отравленных данных. Выполнен анализ динамики изменения метрик качества регрессионной модели в сценарии без состязательных атак, сценарии выполнения состязательной атаки и сценарии изоляции отравленных данных. Выполнение состязательной атаки FGSM с максимизацией знака градиента увеличивает значение метрики MSE в среднем на 33% и снижает значение метрики R^2 в среднем на 10%. Бинарный классификатор LightGBM с точностью в 98% успешно обнаруживает записи с состязательными аномалиями в табличных данных. Регрессионные модели машинного обучения уязвимы к состязательным атакам, при этом своевременный интеллектуальный анализ сетевого трафика и передаваемых по сети данных позволяет обнаруживать злонамеренную сетевую активность.*

Научная новизна: *исследованы методы выполнения состязательных атак на регрессионную модель для задачи прогнозирования комбинированных потерь пути распространения сигнала от базовой станции до конечных пользователей в эмулируемом сегменте беспроводных сетей последнего поколения.*

Ключевые слова: *состязательные атаки, беспроводные самоорганизующиеся сети, машинное обучение, регрессия, MIMO.*

RESEARCH ON ADVERSARIAL ATTACKS ON REGRESSION MACHINE LEARNING MODELS IN 5G WIRELESS NETWORKS

Legashev L. V.[3], Zhigalov A. Yu.[4]

The purpose of research: *Study the impact of adversarial attacks on the evaluation metrics of regression ML models.*

The methods of research: *Emulation of signal propagation data in MIMO systems, synthesis of adversarial samples, execution of adversarial attacks on machine learning models, training of binary classifiers to detect adversarial anomalies in data.*

1 Легашев Леонид Вячеславович, кандидат технических наук, ведущий научный сотрудник лаборатории цифровых решений и аналитики больших данных Оренбургского государственного университета, г. Оренбург, Россия. E-mail: silentgir@gmail.com. ORCID: 0000-0001-6351-404X.

2 Жигалов Артур Юрьевич, младший научный сотрудник лаборатории искусственного интеллекта и анализа данных Оренбургского государственного университета, г. Оренбург, Россия. E-mail: leroy137.artur @gmail.com. ORCID: 0000-0003-3208-1629.

3 Leonid V. Legashev, Ph.D., Leading Researcher, Laboratory of Digital Solutions and Big Data Analytics, Orenburg State University, Orenburg, Russia. E-mail: silentgir@gmail.com. ORCID: 0000-0001-6351-404X.

4 Artur Yu. Zhigalov, Junior Researcher, Laboratory of Digital Solutions and Big Data Analytics Orenburg., Orenburg, Russia. E-mail: leroy137.artur@gmail.com. ORCID: 0000-0003-2752-7198

Scientific novelty: *methods for performing adversarial attacks on a regression model for the problem of predicting the combined losses of the signal propagation path from the base station to end users in the emulated segment of the latest generation wireless networks have been studied.*

The result of research: *Scenario generation and exploratory analysis of a dataset using the DeepMIMO emulator carried out. An adversarial attack with gradient sign maximization using the FGSM method was performed. An experimental comparison of binary classifiers for detecting malicious data was performed. An analysis of the dynamics of changes in the evaluation metrics of a regression model was performed in a scenario without adversarial attacks, a scenario under adversarial attack, and a scenario with isolating compromised data. Performing an adversarial FGSM attack with gradient sign maximization increases the value of the MSE metric by an average of 33% and reduces the value of the R^2 metric by an average of 10%. The LightGBM binary classifier successfully detects records with adversarial anomalies in tabular data with 98% accuracy. Regression-based machine learning models are vulnerable to adversarial attacks, but timely intelligent analysis of network traffic and data transmitted over the network can detect malicious network activity.*

Keywords: *adversarial attacks, wireless ad hoc networks, machine learning, regression, MIMO.*

Введение и обзор современного состояния исследований

Повсеместное распространение беспроводных сетей последнего поколения, развитие технологий миллиметровых радиоволн (mmWave), антенных систем massive MIMO (massive Multiple Input Multiple Output) и, как следствие, возросший уровень передаваемых по сети данных от множества пользователей влечет за собой проблемы обеспечения сетевой безопасности. Автор публикации [1] посвящает свою работу анализу безопасности физического уровня для беспроводных сетей 5G/6G. Современные модели машинного обучения (МО) активно используются для анализа сетевого трафика и выявления злонамеренной сетевой активности, но при этом сами модели глубокого обучения могут быть уязвимы к состязательным атакам, цель которых заключается в компрометации эффективности таких моделей. Состязательные атаки вида «белый ящик» (white box attacks) характерны для случаев, в которых злоумышленник имеет прямой доступ к моделям машинного обучения с возможностью исследования исходного кода и архитектуры. Состязательные атаки вида «черный ящик» (black box attacks) характерны для случаев, в которых злоумышленник имеет возможность тестировать готовую модель. Намеренное добавление специально подготовленных состязательных возмущений в исходные данные может привести к компрометации качества модели машинного обучения.

Множество актуальных исследований состязательных атак посвящены проблеме классификации на основе табличных или графических данных, авторы статьи [2] выполняют комплексный анализ состязательных атак на системы машинного обучения и обсуждают методы их защиты. Следует отметить, что практически отсутствуют публикации по исследованию состязательных атак на задачи регрессии, в том числе в области беспроводных сетей, что подчеркивает актуальность настоящего исследования. Авторы публикации [3] выполняют состязательную атаку вида «белый ящик» на табличные данные, успешно обманывая нейронную сеть и снижая её производительность. В исследовании [4] проводится анализ устойчивости сильно параметризованных линейных моделей к состязательным атакам с целью максимизации ошибки прогнозирования. В статье [5] исследуется устойчивость коэффициентов регрессии к состязательным примерам, подготовленным для «отравления» исходных данных обучения модели МО. Публикация [6] посвящена анализу уязвимости регрессионных моделей многомерных временных рядов к состязательным атакам. Авторы показывают, что исследуемые модели уязвимы к проводимым атакам, что критически важно для безопасности. В исследовании [7] представлены два алгоритма для выполнения состязательных атак на модели регрессии. Авторы статьи [8] отмечают, что подготовленные состязательные примеры, сгенерированные для атаки «белого ящика» можно эффективно использовать для выполнения состязательной атаки на неизвестную злоумышленнику модель регрессии, то есть для выполнения атаки вида «черный ящик». В публикации [9] выполнялось исследование по обнаружению состязательных атак на модели прогнозирования LSTM и временной свёрточной сети на основе алгоритмов одноклассового метода опорных векторов и локального уровня выброса. Авторы статьи [10] описывают общий подход, основанный на анализе возмущений алгоритмов обучения для выполнения состязательных атак на регрессионные модели. В публикации [11] исследуются способы ослабления негативного влияния состязательных примеров на модель робастной непараметрической регрессии. Авторы исследования [12] отмечают важность обеспечения безопасности в автомобильных самоорганизующихся сетях и исследуют различные

варианты выполнения состязательных атак на модели регрессии и варианты защиты от них.

В этой статье будет проведено исследование влияния состязательных атак на метрики качества моделей машинного обучения, а также способы обнаружения таких атак в различных моделируемых сценариях распространения сигнала MIMO антенн.

1. Методы генерации состязательных примеров

Состязательная атака уклонения (dodging attack) в случае задачи классификации является атакой, при которой злоумышленник ставит задачу неправильной классификации объекта, при этом неважно, как именно будет классифицирован объект и к какому некорректному классу он будет отнесен. В случае задачи регрессии атака уклонения заключается в резком увеличении порога ошибки модели регрессии, предсказываемое значение должно быть как можно больше/меньше реального значения. Состязательная отравляющая атака (poisoning attack) – вид атаки, выполняемой в момент обучения моделей искусственного интеллекта, связанных с подмешиванием «отравленных» данных в тренировочный набор данных. В публикации [13] проводится анализ модификаций моделей машинного обучения посредством отравления данных для обучения с количественной оценкой рисков при разработке систем с искусственным интеллектом.

Рассмотрим базовые подходы для генерации состязательных образцов, которые могут быть применены для атаки моделей машинного обучения, построенных на основе табличных данных. Наиболее популярным подходом является использование метода быстрого знака градиента.

1.1. Алгоритм быстрого знака градиента (Fast Gradient Sign Method, FGSM)

Идея данного метода заключается в том, что он вычисляет градиенты функции потерь по отношению к исходным данным, а затем использует знак градиентов для создания нового «отравленного» изображения, которое максимизирует потери J модели машинного обучения:

$$x' = \varepsilon \cdot sign(\nabla_x J(\theta,x,y)), \quad (1)$$

где ε – минимальный уровень шума, θ – модель нейронной сети, $sign(\nabla_x J(\theta,x,y))$ – знак градиента, ∇_x – градиент, x – исходные данные, y – целевое значение для x.

Также следует отметить следующие алгоритмы атак на табличные данные:

1.2. Алгоритм атаки на расстоянии (Distance-based attack)

Данный метод состоит в том, чтобы минимизировать расстояние между объектом и синтетической записью с разными выходными метками. Особенность данного подхода состоит в предварительной группировке состязательных образцов в соответствии с квазиидентификаторами и выставлении соответствующего секретного признака как наиболее распространенное значение (моду). Для данного алгоритма правило обновления можно задать следующим образом:

$$y' = argmax_{|y'i-y|}\ min_{r\in R} \|(x'_i \mid t) - r\|_2, \quad (2)$$

где r – вектор возмущений значений признаков.

1.3. Алгоритм низкого профиля (Low Profile Algorithm)

Данный метод [14] состоит в том, чтобы минимизировать взвешенную норму вектора возмущений на признаках табличных данных при максимизации доли примеров $x \in X$, с ложными ответами на выходе. Для данного алгоритма правило обновления можно задать следующим образом:

$$x'_{i+1} = Clip\{x' + (r'_i + \alpha \cdot \\ \cdot [-\nabla r_j(x'_i,t) + \lambda \|v \odot r\|_p]\}, i = 0,\ldots,N-1, \quad (3)$$

$$x' = argmin_{x'_i}\ d_v(x_i)$$

где λ – коэффициент компромисса, v – вектор важности признаков, N – максимальное количество итераций, α – коэффициент масштабирования

2. Генерация и исследование наборов данных массивных MIMO сетей

2.1. Генерация набора данных сценария «Boston5G_28»

Для генерации наборов данных массивных MIMO сетей на основе точной 3D-трассировки лучей Remcom мы использовали фреймворк DeepMIMO [15]. Рассмотрен сценарий «Boston5G_28» – сценарий на открытом пространстве, созданный на основе центра Бостона, со зданиями варьируемой высоты. На улице зафиксирована одна базовая станция (BS) на высоте 15 м, оборудованная всенаправленной антенной. В качестве массивов пользователей (UE) выступают две сетки антенн с общим количеством в 965 090 пользователей, расположенных на высоте 2 м, расстояние между пользователями – 37 см. Стандартная рабочая частота эмуляции – 28 ГГц. Каждый пользователь состоит из одной всенаправленной антенны. Расстояние между углами трассировки лучей составляет 0,25 градусов. Бетон и влажная земля используются в качестве материалов для зданий и местности соответственно. Модель распространения сигнала такова, что каждый путь канала может пройти максимум через 4 отражения, прежде чем сигнал базовой станции достигнет приемника (пользователя). Задана пропускная способность в 0.1 МГц. Общая схема расположения пользователей и базовой станции представлена на рисунке 1. Бóльшая часть пользователей отрезана от базовой станции

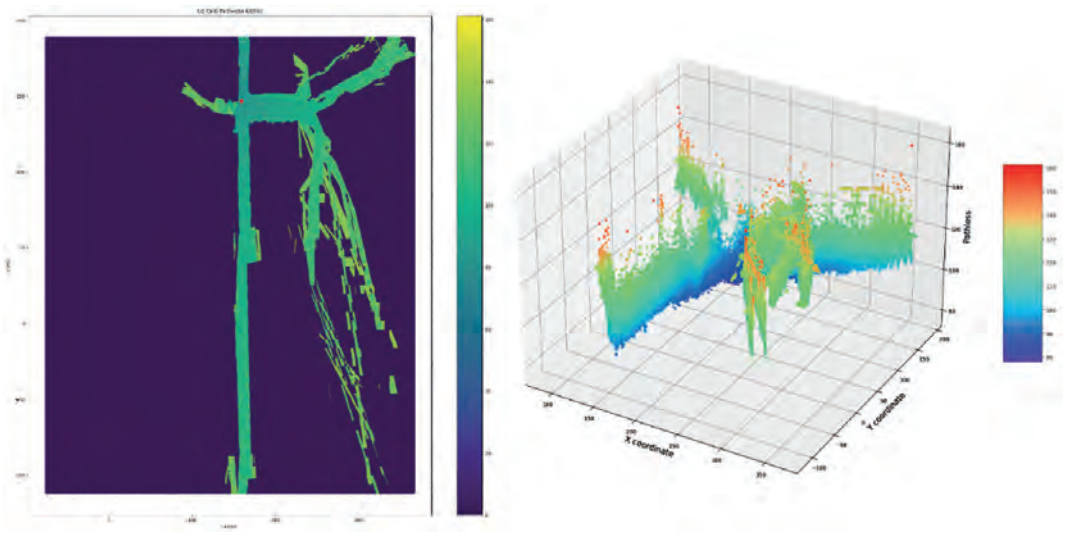


Рис.1. 2d- и 3d- карты городского пространства для сценария «Boston5G_28». Базовая станция отмечена красной точкой на 2d-карте. Цветовая схема соответствует комбинированным потерям сигнала на пути канала между пользователями и базовой станцией. Желтый/красный цвет – высокие потери, зеленый цвет – средние потери, темно-синий цвет – низкие.

в соответствии с топологией эмулируемого сегмента города. Можно динамически отслеживать изменение комбинированных потерь сигнала на пути распространения от источника (базовой станции) до конечных пользователей с учётом архитектуры эмулируемого сегмента города и отражений сигнала.

Для сгенерированного набора данных доступны координаты отправителя и получателей, матрица каналов отправителя и получателей, а также различные числовые характеристики распространении сигнала. В итоговый набор данных после выполнения расчетов сценария выбраны следующие признаки:

1. ***X coordinate*** – координата на оси X пользователя относительно эмулируемой области.
2. ***Y coordinate*** – координата на оси Y пользователя относительно эмулируемой области.
3. ***Distance*** – расстояние между базовой станцией и каждым пользователем, в метрах.
4. ***Pathloss*** – комбинированные потери на пути канала между отправителем и получателем («затухание» сигнала антенны), в децибелах относительно 1 милливатта.
5. ***DoA_phi*** – азимутальный угол прибытия, в градусах.
6. ***DoA_theta*** – зенитный угол прибытия, в градусах.
7. ***DoD_phi*** – азимутальный угол отправления, в градусах.
8. ***DoD_theta*** – зенитный угол отправления, в градусах.
9. ***Phase*** – фаза пути распространения сигнала, в градусах.
10. ***Power*** – сила сигнала при получении, в ватт.
11. ***Time of arrival*** – время получения сигнала, в секундах.
12. ***Line of Sight (LoS)*** – статус сигнала, принимаемый одно из трёх значений из {-1, 0, 1}.

($LoS = 1$): Путь прямой видимости существует. ($LoS = 0$): существуют только пути вне прямой видимости, при этом путь прямой видимости заблокирован. ($LoS = -1$): между передатчиком и приемником нет путей (полная блокировка).

2.2. Исследование сгенерированного набора данных

Итоговый набор данных содержит 105 842 записей, при этом 40 387 пользователей находятся в зоне прямой видимости базовой станции ($LoS = 1$), а 65 455 пользователей находятся вне зоны прямой видимости базовой станции ($LoS = 0$). Метрика ***pathloss*** – комбинированные потери на пути канала – является одной из ключевых метрик оценки качества беспроводных сетей последнего поколения и указывает насколько эффективной является действующая сетевая топология. Значение ***pathloss*** можно прогнозировать на основе имеющихся данных о состоянии сети при передаче сигнала между базовой станцией и большим массивом пользователей. Состязательная атака на модель регрессии должна резко увеличивать или уменьшать предсказываемое значения по отношению к оригинальному значению целевого столбца. Для выполнения состязательной атаки на модель прогнозирования потерь сигнала злоумышленнику выгодно резко увеличивать прогнозируемое значение. Злоумышленники могут выполнять атаку на регрессионные модели машинного обучения путем отравления исходных данных, в результате чего комбинированные потери на пути канала резко

возрастут, и пользователи могут потерять доступ к базовой станции в соответствии с действующими протоколами маршрутизации. В текущем исследовании сфокусируемся на задаче генерации, обнаружения и противодействия таким состязательным атакам.

На рисунке 2 представлены графики рассеяния и гистограммы для комбинированных потерь на пути канала, а также значимость признаков для прогнозирования.

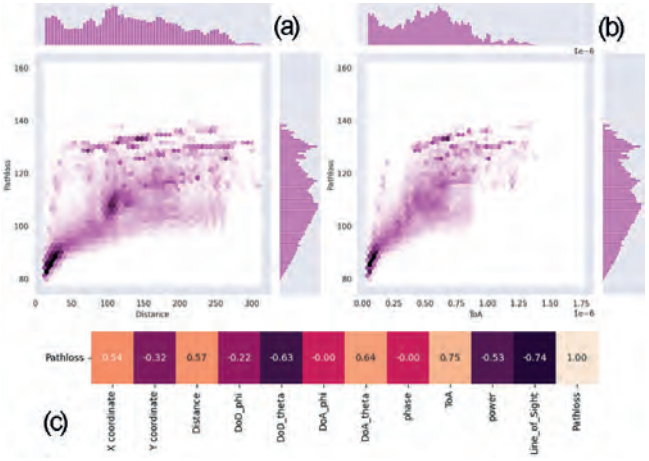


Рис. 2. Гистограммы распределения признака Pathloss в зависимости от расстояния пользователя до базовой станции (a) и в зависимости от времени прибытия сигнала (b), фрагмент матрицы корреляции (c).

Из рисунков 2(a) и 2(b) мы можем визуально выделить три пика высоких потерь сигнала в зависимости от расстояния пользователя до базовой станции и в зависимости от времени прибытия сигнала. На рисунке 2(c) представлен фрагмент матрицы корреляции, показывающий сильную прямую зависимость признака ***Pathloss*** от признаков ***Time of arrival***, ***DoA_theta*** и ***Distance*** и сильную обратную зависимость от признаков ***Line of sight***, ***DoA_theta*** и ***Power***. Действительно, увеличение времени получения сигнала приводит к увеличению комбинированных потерь на пути канала. Для пользователей, находящихся в зоне прямой видимости базовой станции, комбинированные потери на пути канала уменьшаются ввиду отсутствия отражений сигнала по пути его распространения.

3. Исследование состязательных атак на регрессионные модели машинного обучения

Полученный в разделе 2 набор данных разбит в соотношении 40:40:20 на тренировочную выборку для обучения регрессионной модели, выборку для отравления данных при выполнении состязательной атаки и тестовую выборку для валидации данных. Варьирование всех элементов знака градиента позволяет контролировать «направление» ошибки. На рисунке 3 показано, как изменяется прогнозируемое значение комбинированных потерь сигнала pathloss в зависимости от знака градиента.

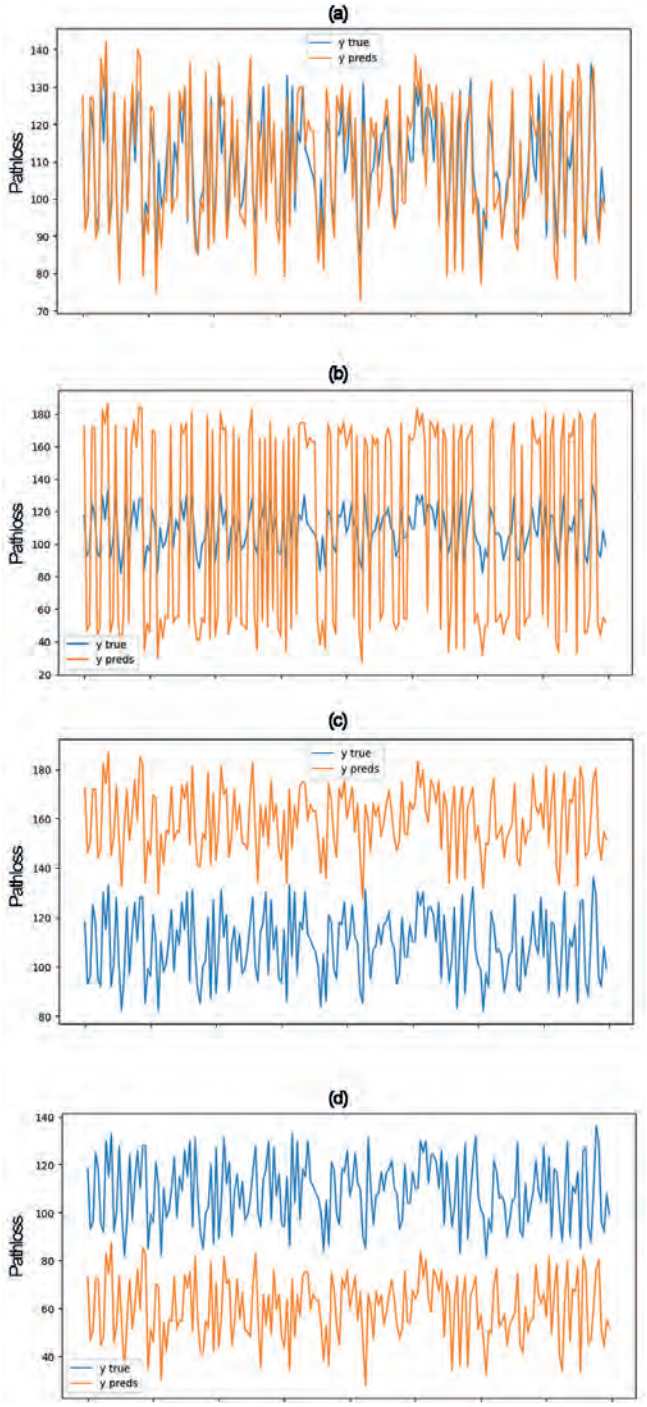


Рис. 3. Фрагмент тестового набора данных в различных сценариях: (a) – обученная модель линейной регрессии, (b) – атака FGSM с флуктуацией знака градиента, (c) – атака FGSM с максимизацией знака градиента, (d) – атака FGSM с минимизацией знака градиента

В текущем исследовании рассмотрены три основных сценария исследования состязательных атак на табличные данные:

1. Сценарий обучения модели регрессии без сторонних вмешательств (Undefended Model). Выполним обучение регрессора LinearRegressor для задачи прогнозирования комбинированных потерь ***pathloss*** по метрике оценки качества Mean Squared Error (MSE) и R^2. Линейная регрессионная модель показала хорошую точность (см. рисунок 4(а)) при решении задачи прогнозирования показателя ***pathloss*** на основе других признаков.

При построении архитектуры нейронной сети градиентный спуск сходится в локальной точке экстремума, поэтому общий алгоритм обучения модели регрессии выглядит следующим образом:

1.1 Выполнено обучение линейной регрессии из библиотеки sklearn на основе метода наименьших квадратов.

1.2 Полученные веса и свободный коэффициент (сдвиг) использованы при инициализации нейронной сети с одним линейным слоем и без функции активации с помощью библиотеки pytorch.

1.3 Выполнено тестирование построенной нейронной сети.

1.4 Подсчитаны метрики качества регрессионной модели.

2. Сценарий отравления исходных данных для обучения на основе генеративно-состязательных сетей (Attacked Model). Выполним состязательную атаку FGSM с варьированием показателя окрестности $\varepsilon = [1^{-10}, 1^{-9}, 1^{-8}, 1^{-7}]$ и доли атакуемых данных $fract = [0.2, 0.4, 0.6, 0.8, 0.95, 0.99999]$.

На рисунке 4 показана зависимость метрик качества от размера ε окрестности для обученной модели линейной регрессии. Значение $\varepsilon = 1^{-7}$ и выше приводит к резкому росту значений метрики MSE и снижению значений метрики R^2, что является нецелесообразным при проведении состязательной атаки, т.к. очень сильное отклонение модели будет расцениваться как выброс (outlier) или аномалия в данных.

В результате исследований по варьированию параметров FGSM можем сделать вывод о том, что модель линейной регрессии наиболее уязвима к атаке FGSM с максимизацией знака градиента с параметрами ε = 1^{-10} и fract = 0.99999, в остальных конфигурациях отклонения в метриках незначительны.

3. Сценарий обнаружения и противодействия состязательным атакам на исходные данные (Secured Model). Полученные во втором сценарии отравленные наборы данных использованы для обучения классификаторов LightGBM, CatBoost и XGBoost для решения задачи бинарной классификации: обычные данные (benign data) с меткой «0» или отравленные данные (malicious data) с меткой «1». Оптимальные параметры классификаторов подобраны с помощью

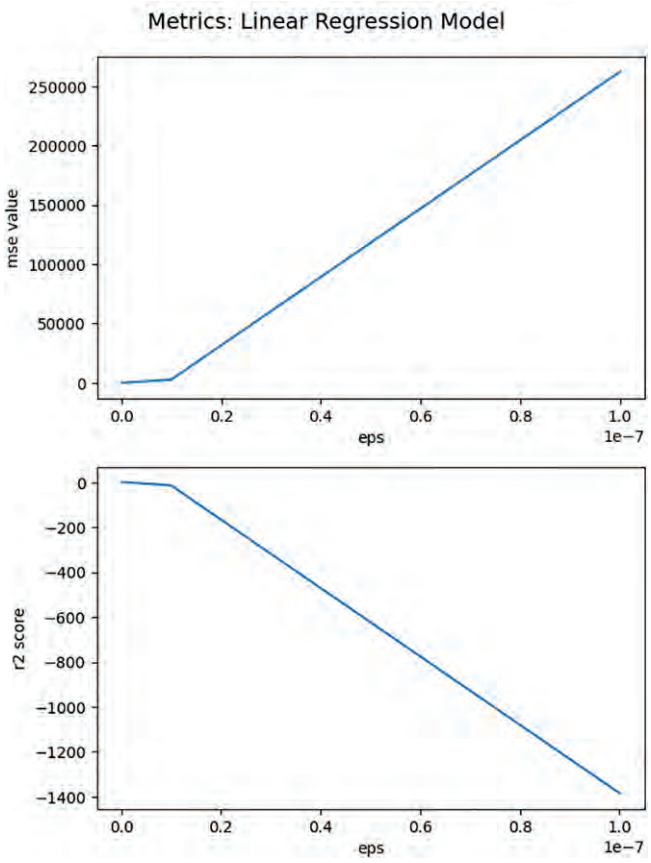


Рис. 4. Зависимость значений метрик MSE и R^2 score от размера ε окрестности

инструмента GridSearchCV. В таблице 1 представлены результаты сравнения трёх классификаторов. Для обучения классификаторов случайным образом выбран набор данных, в котором отравленные и обычные данные пропорционально сбалансированы.

Таблица 1

Сравнение бинарных классификаторов по обнаружению аномалий

Классификатор	$\varepsilon = 1^{-10}$, $fract = 0.6$		
	Precision	**Recall**	**F1-score**
LGBMClassifier (max_depth=20, n_estimators=500, num_leaves=20, subsample=0.7)	0.9835	0.9833	0.9834
CatBoost (depth'=4, 'learning_rate'=0.02, 'iterations'=100)	0.9777	0.9646	0.9703
XGBoost (n_estimators=500)	0.9828	0.9816	0.9822

Лучшие результаты по обнаружению состязательных аномалий показывает классификатор LightGBM с параметрами max_depth=20, n_estimators=500, num_leaves=20, и subsample=0.7. По результатам

работы классификатора на тестовых данных удалим из набора данных обнаруженные состязательные примеры и получим сокращенный набор данных из 5029 записей, на котором повторно выполним оценку качества регрессионной модели.

4. Обсуждение и выводы

На каждом из трёх этапов выполнялся подсчет основных метрик качества регрессионных моделей. В таблице 2 представлена динамика изменения метрик качества линейной регрессионной модели в зависимости от исследуемого сценария. Выполнение состязательной атаки FGSM с максимизацией знака градиента и параметрами показателя окрестности $\varepsilon = 1^{-10}$ и доли атакуемых данных $fract = 0.99999$ увеличивает значение метрики MSE в среднем на 33% и снижает значение метрики R^2 в среднем на 10%. Бинарный классификатор LightGBM с подобранными оптимальными гиперпараметрами с точностью в 98% успешно обнаруживает записи с состязательными аномалиями в табличных данных, изоляция которых позволяет восстановить метрики регрессионной модели до исходных значений.

В рамках проведенного исследования выполнена генерация табличных данных сценария сегмента беспроводной сети на базе эмулятора DeepMIMO; выполнено построение состязательных примеров с целью максимизации прогнозируемого значения комбинированных потерь сигнала от базовой станции до конечных пользователей; выполнено обучение бинарного классификатора по распознаванию отравленных данных; показана динамика изменения метрик качества линейной регрессионной модели в приложениях беспроводных сетей поколения 6G. Регрессионные модели машинного обучения уязвимы к состязательным атакам, своевременный интеллектуальный анализ сетевого трафика и передаваемых по сети данных может обнаруживать злонамеренную сетевую активность в сегменте беспроводной сети последнего поколения.

Таблица 2

Динамика изменения метрик качества линейной регрессионной модели

Сценарий	$\varepsilon = 1^{-10}$, $fract = 0.99999$	
	MSE	R2
Undefended Model {Linear Regression}	38.51	0.80
Attacked Model {FGSM}	51.40 ↑	0.72 ↓
Secured Model {LightGBM}	37.55 ↓	0.80 ↑

Исследование выполнено за счет гранта Российского научного фонда (проект № 22-71-10124).

Литература

1. *Петров И. А. Безопасность физического уровня для сетей 5G/6G // Вопросы кибербезопасности. – 2023. – №. 3. – С. 55.*
2. *Котенко И. В. и др. Атаки и методы защиты в системах машинного обучения: анализ современных исследований // Вопросы кибербезопасности. – 2024. – №. 1. – С. 59.*
3. *Gupta K. et al. An adversarial attacker for neural networks in regression problems // IJCAI Workshop on Artificial Intelligence Safety (AI Safety). – 2021.*
4. *Ribeiro A. H., Schön T. B. Overparameterized linear regression under adversarial attacks // IEEE Transactions on Signal Processing. – 2023. – V. 71. – P. 601–614.*
5. *Li F., Lai L., Cui S. On the adversarial robustness of linear regression // 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP). – IEEE, 2020. – P. 1–6.*
6. *Mode G. R., Hoque K. A. Adversarial examples in deep learning for multivariate time series regression // 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR). – IEEE, 2020. – P. 1–10.*
7. *Kong X., Ge Z. Adversarial Attacks on Regression Systems via Gradient Optimization // IEEE Transactions on Systems, Man, and Cybernetics: Systems. – 2023.*
8. *Meng L. et al. White-box target attack for EEG-based BCI regression problems // Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part I 26. – Springer International Publishing, 2019. – P. 476–488.*
9. *Santana E. J. et al. Detecting and mitigating adversarial examples in regression tasks: A photovoltaic power generation forecasting case study // Information. – 2021. – V. 12. – №. 10. – P. 394.*
10. *Balda E. R., Behboodi A., Mathar R. Perturbation analysis of learning algorithms: Generation of adversarial examples from classification to regression // IEEE Transactions on Signal Processing. – 2019. – V. 67. – №. 23. – P. 6078–6091.*
11. *Zhao P., Wan Z. Robust nonparametric regression under poisoning attack // Proceedings of the AAAI Conference on Artificial Intelligence. – 2024. – V. 38. – №. 15. – P. 17007–17015.*
12. *Deng Y. et al. An analysis of adversarial attacks and defenses on autonomous driving models // 2020 IEEE international conference on pervasive computing and communications (PerCom). – IEEE, 2020. – P. 1–10.*
13. *Костогрызов А. И., Нистратов А. А. Анализ угроз злоумышленной модификации модели машинного обучения для систем с искусственным интелектом // Вопросы кибербезопасности. – 2023. – №. 5. – С. 9.*
14. *Ballet V. et al. Imperceptible adversarial attacks on tabular data // arXiv preprint arXiv:1911.03274. – 2019. DOI: https://doi.org/10.48550/arXiv.1911.03274*
15. *Alkhateeb A. DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications // arXiv preprint arXiv:1902.06435. – 2019. DOI: https://doi.org/10.48550/arXiv.1902.06435*

МЕТОДИКА РАЗРАБОТКИ АВТОМАТИЗИРОВАННЫХ СРЕДСТВ ГЕНЕРАЦИИ ПРОГРАММНОГО КОДА ПОСРЕДСТВОМ НАСТРОЙКИ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ

Самонов А. В.[1], Бурова И. О.[2]

DOI: 10.21681/2311-3456-2024-3-68-75

Цель исследования: *разработка методического, алгоритмического и программного обеспечения для создания автоматизированных средств генерации программного обеспечения на основе больших языковых моделей.*

Методы исследования: *анализ архитектуры, методов и средств создания, обучения и применения больших языковых моделей, исследование методов и алгоритмов точной настройки и применения больших языковых моделей для генерации программного кода, экспериментальные исследования разработанных алгоритмов и программ на стенде.*

Полученные результаты: *исследованы архитектурные и технологические основы построения и функционирования больших языковых моделей (Large Language Model, LLM). Определены перспективные технологии, методы и средства обучения и точной настройки LLM на решение задач в области программирования. Разработана методика создания автоматизированных средств генерации программного кода посредством реализации итерационной процедуры настройки ограниченного количества значимых параметров базовой LLM на специально подготовленных обучающих наборах данных. Определены ключевые модули и параметры процедуры настройки LLM. Представлены фрагменты программной реализации методики в среде Pytorch. Полученные в ходе экспериментов результаты свидетельствуют о целесообразности применения данного подхода для разработки автоматизированных средств генерации программного кода.*

Научная и практическая значимость: *состоит в разработке методического, алгоритмического и программного обеспечения, предназначенного для создания при ограниченных вычислительных ресурсах на основе больших языков моделей автоматических средств генерации и тестирования программного кода, в которых отсутствуют катастрофическое забывание, риск переобучения, галлюцинации.*

Ключевые слова: *большие языковые модели, глубокое обучение, внимание на себя, нейросетевые модели, трансформер, Large Language Model, self-attention, transformer*

METHODOLOGY FOR THE DEVELOPMENT OF AUTOMATED SOFTWARE CODE GENERATION TOOLS BY FINE-TUNING LARGE LANGUAGE MODELS

Samonov A. V.[3], Burova I. O.[4]

The purpose of research: *the development of methodological, algorithmic and software for the creation of automated software generation tools based on large language models*

Research methods: *analysis of architecture, methods and means of creating, teaching and applying large language models, research of methods and algorithms for fine-tuning and applying large language models to generate program code, experimental studies of developed algorithms and programs on the stand.*

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

2 Бурова Ирина Олеговна младший научный сотрудник, Военно-космическая академия имени А. Ф. Можайского, Санкт-Петербург, Россия, E-mail: burova@smilecom.ru

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

4 Irina O. Burova. research scientist Mozhaiskiy Military Space Academy St.Petersburg, Russia. E mail: burova@smilecom.ru

The results obtained: *the architectural and technological foundations of the construction and functioning of large language models (LLM) are investigated. Promising technologies, methods and tools for teaching and fine-tuning LLM to solve programming problems have been identified. A methodology has been developed for creating automated software code generation tools by implementing an iterative procedure for configuring a limited number of significant parameters of the basic LLM on specially prepared training datasets. The key modules and parameters of the LLM setup procedure are defined. Fragments of the software implementation of the technique in the Pytorch environment are presented. The results obtained during the experiments indicate the expediency of using this approach to develop automated software code generation tools.*

Scientific and practical significance: *it consists in the development of methodological, algorithmic and software designed to create, with limited computing resources, models of automatic means of generating and testing software code based on large languages models, in which there is no catastrophic forgetting, the risk of retraining, hallucinations.*

Keywords: *large language models, deep learning, neural network models, transformer, Large Language Model, self-attention.*

Введение

Современный этап мирового развития характеризуется активным внедрением в индустрию, науку, образование и другие сферы хозяйственной и общественной жизни технологий искусственного интеллекта. Яркими примерами таких технологий являются методы глубокого обучения и генеративный искусственный интеллект. Созданные с их помощью большие языковые модели (Large Language Model, LLM) и программные комплексы способны обрабатывать и создавать тексты, понимать и синтезировать речь, изображения, музыку, генерировать программный код, решать аналитические, математические и другие нетривиальные задачи. Наиболее известными коммерческими LLM, используемыми в области разработки программного обеспечения, являются: GPT-4, GPT-3.5, Claude 2, Palm 2, AlphaCode 2 [1]. Примерами LLM с открытым исходным кодом, предназначенных для решения задач аналогичного класса, являются: Code Llama, WizardCoder, PhindCodeLlama, StarCoder, CodeGen, CodeGeeX и ряд других [2]. Данные системы могут обнаружить и помочь устранить ошибки в программном коде, предложить вариант решения типовой задачи, а также самостоятельно сгенерировать программный код на таких языках как Python, C++, Java, JavaScript, Go и др. На рис. 1 представлены результаты тестирования возможностей средств генерации программного кода с помощью тестовых наборов HumanEval и MBPP [3]. Как видно из данного графика, целый ряд средств успешно справляются с половиной и более тестовых заданий.

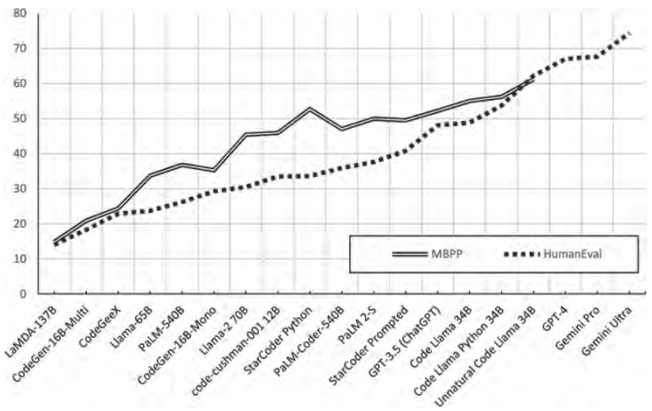


Рис. 1. Результаты сравнительного анализа производительности программных пакетов, используемых для генерации программного кода

Разработка таких средств осуществляется посредством настройки больших языковых моделей на решение задач в области программирования. Как показал проведенный анализ, современные средства генерации программного кода пока способны разрабатывать только относительно простые программы и преимущественно на языке Python. Основными проблемными вопросами, с которыми сталкиваются разработчики систем данного класса, являются: катастрофическое забывание, риск переобучения, галлюцинации создаваемых систем, а также исключительно высокие требования к производительности используемых при обучении LLM вычислительных средств.

В данной статье представлены предложения по совершенствованию и развитию методического, алгоритмического и программного обеспечения процессов создания средств генерации программного кода на основе больших языковых моделей, позволяющие преодолеть существующие трудности и ограничения.

Архитектура, технологии и алгоритмы обучения и функционирования больших языковых моделей

Технологическая цепочка процессов и средств, реализуемых и используемых при создании базовой модели LLM и последующей ее специализации, представлена на рисунке 2. На начальном этапе модель

обучается на неструктурированных и немаркированных данных. Основными источниками данных, которые были использованы при разработке большинства современных LLM, являются: Wikipedia, Common Craw, BooksCorpus (коллекция текстов книг), OpenWebText (набор статей из сети интернет). В результате получается предварительно обученная (pretrained) LLM общего назначения. Примерами таких моделей являются GPT 4, GPT 3.5, Gemini, Falcon, Llama, Mixtral. На втором этапе осуществляется доработка LLM посредством самостоятельного обучения на специальным образом подготовленных и маркированных данных, настраивающая модель на решение задач определенного класса. На третьем этапе такие модели проходят дополнительное обучение с подкреплением на основе обратной связи с экспертом (Reinforcement Learning from Human Feedback). Примерами наборов данных, используемых при разработке средств генерации и тестирования программного кода, являются CodeTextBook, sql-create-context. В результате создаются предметно-ориентированные программные системы ИИ (СИИ), предназначенные для решения конкретных практических задач в определенных областях. При адаптации LLM на решение задач в области программирования используются такие источники как Github и StackOverflow, наборы данных MathQA-Python, MBPP, APPS, DS1000. В результате создается специализированная (обученная и проверенная) СИИ, включающая несколько взаимосвязанных нейросетей и программных средств, обеспечивающих их настройку и применение для решения задач в области программирования.

Рис.2. Технологическая цепочка процессов и средств, используемых при создании СИИ

Бурное развитие и широкое применение LLM во многом обязано двум используемым в них техническим решениям: архитектуре «трансформер» (transformer) и механизму «внимание на себя» (self-attention). Состав, структура и механизмы функционирования основных компонентов трансформера представлены на рисунке 3 [4, 5].

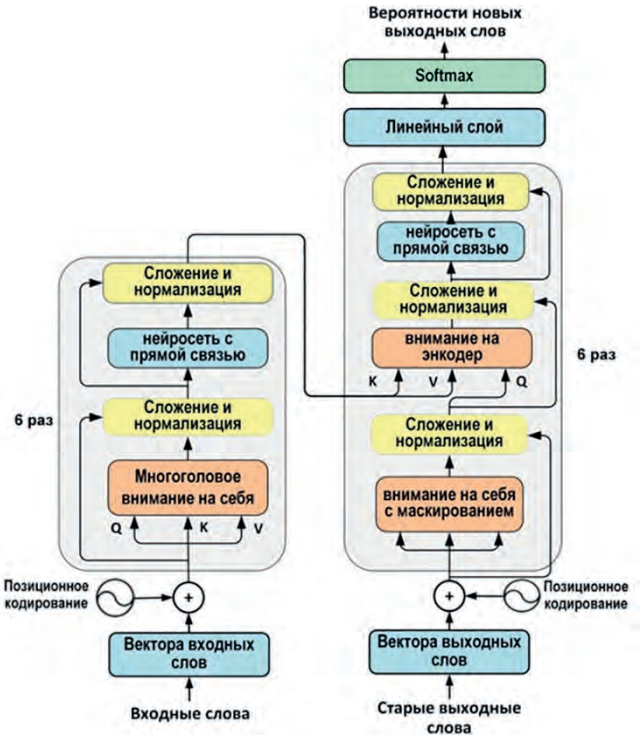


Рис.3. Архитектура трансформера

Устройство трансформера состоит из кодирующего и декодирующего компонентов. На вход принимается некая последовательность, создается ее векторное представление (англ. embedding), прибавляется вектор позиционного кодирования, после чего набор элементов без учета их положения в последовательности поступает в кодирующий компонент (encoder), а затем декодирующий компонент (decoder) получает на вход часть этой последовательности и выход кодирующего. В результате получается новая выходная последовательность. Трансформер-кодировщик переводит исходные векторы в скрытые, которые правильно сохраняют в себе информацию о контексте каждого элемента. Каждый слой энкодера включает следующие модули: внимание на себя (self-attention), сложение и нормализацию (add&normalize), нейросеть прямого распространения (FFN, feed-forward neural network).

Далее трансформер-декодировщик декодирует результат кодировщика в новую последовательность, которая состоит из эмбедингов элементов выходного языка. По эмбедингам генерируются итоговые элементы с помощью вероятностной языковой модели. Результаты работы энкодера принимает модуль декодера «внимание на энкодер». При получении данных модуль декодера формирует запрос Q из данных модуля «внимание на себя с маскированием» и ищет соответствующие ему ключи K и значения V. Модуль «внимание на энкодер» передает работу в нейросеть прямого распространения. Данные проходят через

шесть слоев декодера, которые включают такие же, как и в энкодере модули. С последнего слоя декодера результат попадает на заключительные модули – линейный слой и softmax. Данная процедура выполняется до тех пор, пока входная матрица декодера не заполнится до конца и не сгенерируется сигнал остановки.

Сердцем трансформера является работа модуля «внимание к себе», с помощью которого трансформер определяет контекст обрабатываемого в данный момент слова (токена) и определяет степень его близости с другими словами (токенами) входного набора данных. Первым шагом при вычислении «внимания к себе» является создание из входного вектора трех векторов: $Q(x)$ – запроса, $K(x)$ – ключа и $V(x)$ – значения. Эти векторы создаются путем умножения входного вектора на соответствующие им матрицы: W_Q, W_K и W_V, которые были рассчитаны при обучении LLM [4]:

$$Q_{(x)} = x \cdot W_Q + b_k \quad (1)$$

$$K_{(x)} = x \cdot W_K + b_q \quad (2)$$

$$V_{(x)} = x \cdot W_V + b_v, \quad (3)$$

где W_Q, W_K, $\in \mathrm{R}^{\text{input} \times \text{key}}$, $W_V \in \mathrm{R}^{\text{input} \times \text{val}}$, b_Q, $b_K \in \mathrm{R}^{\text{input} \times \text{key}}$, $b_V \in \mathrm{R}^{\text{input} \times \text{val}}$. (4)

Один набор Q, K и V может отражать только один вид зависимостей между токенами, и матрицы извлекают лишь ограниченный набор информации из входных представлений. Чтобы компенсировать эту неоптимальность, в классическую архитектуру трансформера вместо одного слоя внимания включили несколько параллельных с разными весами. Используя тензорную нотацию, процедуру вычисления «многоголового внимания к себе» (MHA, multi-head attention) можно представить в виде следующих формул [6, 7]:

$$MHA(Q,K,V) = Concat(head_1,\ldots,head_h)\ W^O, \quad (5)$$

где Q, K, V – матрицы запросов, ключей и значений соответственно, а W^O матрица с весовыми коэффициентами без смещения на выходе;

$head_i = Attention(xW_iQ, xW_iK, xW_iV)$;

$Attention\ (Q_i,K_i,V_i) = softmax(\frac{Q_i K_i^T}{\sqrt{dk}})\ V_i$;

W_i^Q, W_i^K, W_i^V – матрицы весов для i-й головы модуля внимания.

Вторая часть трансформера – нейросеть прямого распространения (FFN, feed-forward neural network) представляет собой два обычных полносвязных слоя, применяемых независимо к каждому элементу входной последовательности. FFN берет вектор x (скрытое представление в определенной позиции последовательности) и пропускает его через два изученных линейных преобразования (представленных матрицами W_1 и W_2 и векторами смещения b_1 и b_2). Между двумя линейными преобразованиями применяется функция активации:

$$FFN(x) = act(xW_1 + b_1)\ W_2 + b_2. \quad (6)$$

В качестве функции активации act используются ReLU (Rectified Linear Unit), GELU (Gaussian Error Linear Unit) или SwiGLU [8]. Функция Softmax нормализует оценки, чтобы все они были положительными и в сумме равнялись 1.

В результате создается базовая LLM, включающая несколько взаимосвязанных нейросетей и программных средств. Нейросети представлены матрицами W_Q, W_K, W_V. Программные средства включают функции кодирования, декодирования, многоголосового внимания, сложения, нормализации, линеаризации и др. Далее на основе базовой LLM можно создать специализированную систему, предназначенную для решения предметно-ориентированных задач.

Методы и средства создания специализированных интеллектуальных средств для автоматической генерации программного кода

Настройка и специализация больших языковых моделей требует обновления сотен миллионов и миллиардов параметров и сохранения больших копий весовых коэффициентов для каждой задачи, что приводит к увеличению затрат на хранение, совместное использование и обслуживание моделей. При обучении LLM используется гораздо больше памяти, чем при простом ее размещении на графическом процессоре. Это связано с тем, что во время обучения память используется для следующих компонентов LLM: весов модели, состояний оптимизатора, градиентов, переадресации активаций, сохраненных для вычисления градиента, временных буферов. С целью сокращения этих затрат разработаны и используются методы точной настройки значимых (эффективных) параметров (PEFT, Parameter-Efficient Fine-Tuning). Описание современных методов и средств точной настройки LLM представлено в [9–14].

Методы точной настройки эффективных параметров LLM, в отличие от полной настройки модели, обеспечивают обучение только небольшого набора параметров, которые могут быть подмножеством существующих параметров модели или набором добавленных параметров. Эти методы различаются значимостью параметров, эффективностью использования памяти, скоростью обучения, конечным качеством модели и возможными дополнительными затратами при настройке и использовании по назначению. Методы PEFT позволяют повысить корректность и производительность предварительно обученных языковых моделей при решении задач

из определенной области. Основными преимуществами использования методов PEFT являются: сокращение времени обучения, снижение затрат на вычисления и хранение моделей, снижение риска переобучения, преодоление катастрофического забывания, удобство развертывания и переноса на другие устройства.

Наиболее известными методами тонкой настойки являются: LoRA, Prefix tuning, Prompt tuning и Adapters [9–16]. LoRa (Low-Rank Adaptation) – адаптация низкого ранга фокусируется на изменении весов только определенных слоев и параметров базовой LLM, ориентируясь на те, которые наиболее эффективны (полезны) для решения задач данного класса. Это достигается путем применения для настройки весов матриц, имеющих существенно меньший ранг, чем матрицы базовой модели. LoRa применяется только к матрицам запросов и значений трансформера, что означает, что многослойный персептрон заморожен и адаптируются только веса внимания. Функция потерь оптимизируется путем передачи градиента через замороженную модель в адаптеры. Метод QLoRA (Quantization-Aware Low-Rank Adaptation) разработан для развертывания моделей в средах с ограниченными ресурсами. Он позволяет значительно снизить требования к необходимым для развертывания и настройки моделей объемам и производительности памяти GPU и CPU, а также мощностям вычислителей. На рисунке 4 представлена упрощенная схема трансформера (слева) с включенным в него адаптером, реализующим метод LoRA (справа).

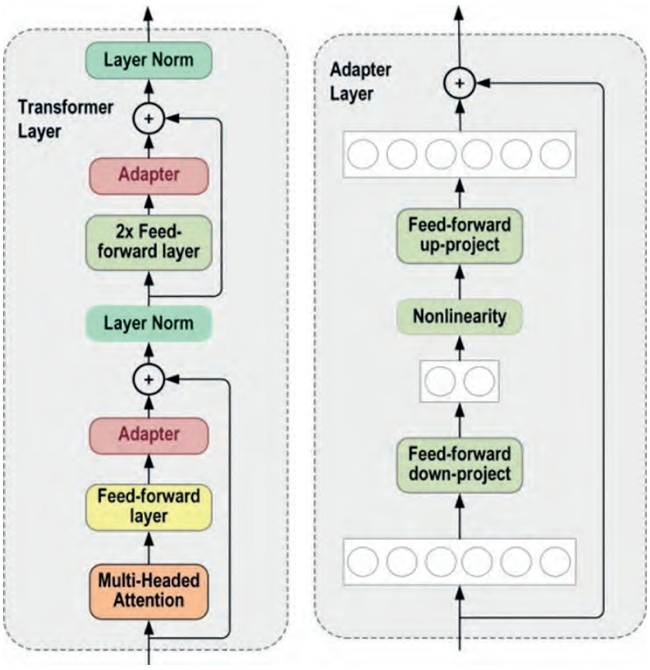


Рис.4. Схема трансформера (слева) с включенным в него адаптером (справа), реализующим метод LoRA

Как показано на рис. 5, LoRa обучает только матрицы A и B, оставляя предварительно обученные веса замороженными [9].

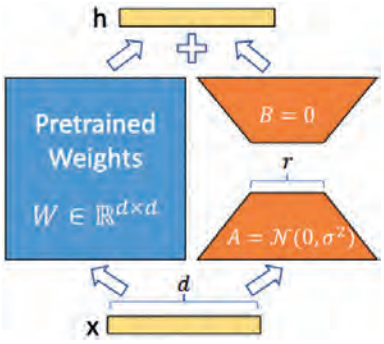


Рис.5. Схема, иллюстрирующая метод LoRA

LoRa позволяет использовать одну и ту же модель для разных задач путем замены весов LoRa, сокращая объем памяти, необходимый для хранения разных моделей. Обучение с помощью LoRa происходит быстрее, поскольку оптимизируются только матрицы LoRa, в отличие от полной точной настройки.

Формула, описывающая в тензорной нотации метод LoRA, имеет следующий вид:

$$W_0 + \Delta W = W_0 + BA,$$

где W_0 – матрица весов предварительно обученной модели; ΔW – обновленные и добавленные весовые коэффициенты во время адаптации исходной модели; r – ранг матрицы с обновляемыми параметрами; $A \in Rr \times k$ – матрица размера rxk, элементами которой являются случайные величины, соответствующие нормальному закону распределения $N(\mu, b^2)$, где μ=0 (среднее значение величины), b (сигма) – среднеквадратическое отклонение; $B \in Rd \times r$ – матрица размера dxr, элементам которой на начальном этапе обучения присваиваются нули.

Важным достоинством LoRa является возможность использовать одну и ту же модель для разных задач путем замены весов в матрицах A и B, сокращая объем памяти, необходимый для хранения разных моделей. QLoRA расширяет метод LoRA посредством квантизации параметров модели, т.е. уменьшения точности весовых коэффициентов, сохраняя при этом необходимую корректность и производительность.

Для специализации LLM в области программирования можно использовать следующие обучающие наборы данных: MBPP, MathQA-Python, MultiPL-MBPP, APPS, DS1000 [17–19]. В результате создается специализированная (обученная и проверенная) СИИ, позволяющая самостоятельно сгенерировать программный код для решения сформулированной на естественном языке задачи или проверить программный код на наличие ошибок и дефектов.

Апробация методики в среде фрэймворка Pytorh

Схема алгоритма, реализующего процедуру адаптации и настройки базовой LLM для ее использования в качестве средства генерации программного кода, представлена на рисунке 6. Для успешного решения данной задачи требуются высокая производительность вычислительных средств и значительные объемы оперативной памяти GPU и CPU. Это обусловлено большими размерами LLM. Например, для запуска Llama 2 7B требуется GPU объемом 13 ГБ, а для ее точной настройки потребуется около 70 ГБ памяти графического процессора. В связи с этим, на первом шаге осуществляется сокращение размера базовой LLM посредством операции квантизации. В результате квантизации на 4 бита для обучения модели потребуется GPU объемом 24 ГБ. При этом потери точности результата работы настроенной модели не превысят 4 процентов. Ниже представлен фрагмент программы, реализующий данную операцию с помощью класса BitsAndBytesConfig фрэймворка Pytorh.

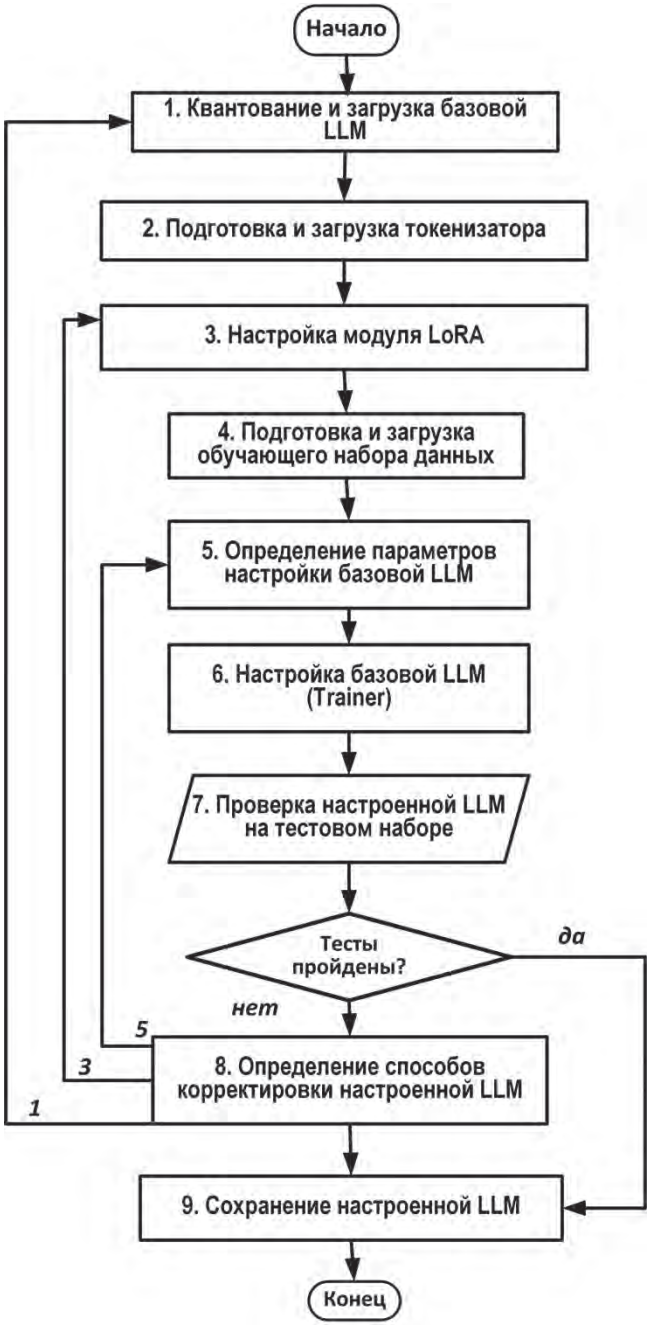


Рис.6. Схема алгоритма, реализующего процедуру точной настройки LLM

```
bnb_config = BitsAndBytesConfig(load_in_4bit=True, bnb_4bit_use_double_quant=True, bnb_4bit_quant_type="nf4", bnb_4bit_compute_dtype=torch.bfloat16)
model = AutoModelForCausalLM.from_pretrained(base_model, device_map="auto", trust_remote_code=True, quantization_config=bnb_config)
```

В результате выполнения данного фрагмента программы линейные слои модели сначала преобразуются в формат fp4/nf4 (`load_in_4bit=True`), а затем выполняется повторная квантизация уже квантизированных весов (`bnb_4bit_use_double_quant=True`).

На втором шаге методики осуществляется загрузка токенизатора настраиваемой LLM:

```
tokenizer = AutoTokenizer.from_pretrained(base_model)
tokenizer.pad_token = tokenizer.eos_token
```

На третьем шаге методики осуществляется настройка модуля LoRA. Пример конфигурационного файла представлен ниже:

```
config = LoraConfig(r=8, lora_alpha=32, target_modules = ["q_proj", "k_proj", "v_proj", "o_proj"], lora_dropout = 0.05, bias="none", task_type="CAUSAL_LM")
```

Основными параметрами являются: r – целое число, определяющее способ обновления матриц, более низкий ранг приводит к менее поддающимся обучению параметрам; `lora_alpha` – коэффициент масштабирования; `lora_dropout` – вероятность выпадения слоев; `task_type` – задает тип используемой модели.

На четвертом шаге осуществляется подготовка и загрузка обучающего набора данных:

```
train_dataset = load_dataset('json', data_files = 'train_set.jsonl', split = 'train')
```

На пятом шаге формируется конфигурационный файл для программы ***trainer***, осуществляющей непосредственное обучение модели. Пример содержимого конфигурационного файла представлен ниже:

```
training_args = transformers.
TrainingArguments (per_device_train_
batch_size = 1, gradient_accumulation_
steps = 8, num_train_epochs=4,
learning_rate=2e-4, fp16=True, save_
total_limit=3, logging_steps=1, output_
dir="experiments", optim ="paged_
adamw_8bit", lr_scheduler_type=
"cosine", warmup_ratio=0.05)
```

Самыми важными являются следующие параметры:

`gradient_accumulation_steps` – количество шагов обновления для накопления градиентов перед выполнением обратного хода/обновления;

`optim="paged_adamw_32bit"`, – используемый оптимизатор;

`learning_rate` – начальная скорость обучения для AdamW optimizer.

На следующем шестом шаге осуществляется настройка базовой модели с помощью модуля `trainer`. В приведенном ниже тексте программы определены базовая модель, обучающий набор данных, конфигурационный файл, функция, используемая для формирования пакета из списка элементов `train_dataset`.

```
trainer = transformers.Trainer(model=
base_model, train_dataset = data,
args = training_args, data_collator =
transformers.DataCollatorForLanguageMo
deling(tokenizer, mlm=False))
```

На седьмом шаге выполняется тестирование полученной в результате обучения специализированной модели. Для оценки качества используются следующие показатели: полнота и корректность выполнения тестовых заданий, использованные вычислительные ресурсы и затраченное время. На основе анализа результатов тестирования принимается решение о прекращении процесса настройки модели, или о продолжении ее обучения и оптимизации. Во втором случае определяется – какие модули и каким образом следует скорректировать, чтобы получить оптимальный результат. В зависимости от выбранного способа коррекции процедура настройки повторяется, начиная с шагов 1, 3 или 5.

Завершается данный процесс, когда тестовые испытания прошли успешно и характеристики производительности стабилизировались на достаточно высоком уровне. Разработанный комплекс нейросетевых моделей и программного обеспечения сохраняется для дальнейшего использования по назначению с помощью следующих операций:

```
trainer.model.save_pretrained(new_
model)
trainer.tokenizer.save_pretrained
(new_model)
```

В качестве базовой LLM для апробации представленной выше методики и реализующих ее программных средств была выбрана LLM Mixtral 8x7B [20]. Выбор данной LLM обусловлен следующими обстоятельствами. Во-первых, модель имеет небольшой размер. Во-вторых, доступна по лицензии Apache 2.0. В третьих, Mixtral 8x7B по многим тестам превосходит или близка к результатам таких моделей как Llama 2 13B и CodeLlama 7B. Отличительной особенностью Mixtral 8x7B является ее архитектура, представляющая собой сеть с разреженной смесью экспертов (SMoE,Sparse Mixture-of-Experts). Mixtral имеет 46,7 млрд общих параметров, но использует только 12,9 млрд параметров для каждого токена. В таблице 1 представлены результаты тестирования возможностей трех LLM: GPT 3.5, Llama 2 и Mixtral8x7B.

Таблица 1

Результаты сравнения возможностей Mixtral с моделями семейства Llama 2 и базовой моделью GPT 3.5

Метод и средства тестирования	GPT-3.5	LLaMA 2 70B	Mixtral 8x7B
MMLU (MCQ in 57 subject)	70.0%	69.9%	70.6%
MBPP (pass@1)	52.2%	49.8%	60.7%
GSM-8K (5-shot)	57.1%	53.6%	58.4%
MT Bench (for Instruct Model)	8.32	6.86	8.30

Представленные в таблице данные свидетельствую о том, что Mixtral 8x7B соответствует или превосходит Llama 2 70B и GPT 3.5 по большинству показателей.

Заключение

В данной статье представлены методы и средства, предназначенные для разработки средств автоматической генерации программного кода, обеспечивающего решение задачи, сформулированной на естественном языке. Основными проблемными вопросами, с которыми сталкиваются разработчики систем данного класса, являются: катастрофическое забывание, риск переобучения, а также исключительно высокие требования

к производительности используемых при этом вычислительных средств – GPU и CPU. Исследование и анализ существующих и перспективных технологий и средств обучения и применения LLM позволил определить наиболее перспективные пути и методы решения и преодоления имеющих проблем и ограничений. В качестве таковых предложено использовать: методы квантизации, точной настройки (LoRA, QLoRA), оптимизации процесса обучения (AdamW, AdaMix) и др. Интегрирующая эти методы и средства методика создания автоматизированных средств генерации программного кода представлена в виде итерационной процедуры настройки (дообучения) ограниченного количества значимых параметров базовой LLM на специально подготовленных наборах данных. Для апробации методики разработана программная реализация в среде Pytorch. Полученные в ходе ее тестирования и применения результаты свидетельствуют о целесообразности применения данного подхода для разработки автоматизированных средств генерации программного кода.

Литература

1. *A Survey of Large Language Models. Wayne Xin Zhao, Kun Zhou, Junyi Li et al. arXiv:2303.18223v13 [cs.CL] 24 Nov 2023.*
2. *Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning https://arXiv:2303.15647v1 [cs.CL] 28 Mar 2023.*
3. *Ankit Yadav, Mayank Singh. Boldly Going Where No Benchmark Has Gone Before: Exposing Bias and Shortcomings in Code Generation Evaluation. arXiv:2401.03855v2 [cs.CL] 23 Feb 2024.*
4. *Attention Is All You Need. Ashish Vaswani, Noam Shazeer, Niki Parmar. arXiv:1706.03762v7 [cs.CL] 2 Aug 2023*
5. *Jay Alammar. The Illustrated Transformer. http://jalammar.github.io/illustrated-transformer.*
6. *Jinjie Ni, Rui Mao, Zonglin Yang. Finding the Pillars of Strength for Multi-Head Attention. arXiv:2305.14380v2 [cs.LG] 15 Oct 2023.*
7. *David Chiang, Alexander M. Rush, and Boaz Barak. 2021. Named tensor notation. ArXiv,abs/2102.13196.*
8. *Noam Shazeer. GLU Variants Improve Transformer. arXiv:2401.03065v1 [cs.SE] 5 Jan 2024.*
9. *LORA: Low-Rank adaptation of large language models. Edward Hu, Yelong Shen, Phillip Wallis and etl., arXiv:2106.09685v2 [cs.CL] 16 Oct 2021.*
10. *LLaMA-Adapter: Efficient Fine-tuning of Language Models with Zero-init Attention. Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao. arXiv:2303.16199v2 [cs.CV] 14 Jun 2023*
11. *Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models. Ning Ding, Yujia Qin, Guang Yang, Fu Wei, et al. ArXiv, abs/2203.06904*
12. *QLoRA: Quantization-aware low-rank adaptation of large language models Yuhui Xu Lingxi Xie Xiaotao Gu Xin Chen Heng Chang arXiv:2309.14717v2 [cs.LG] 9 Oct 2023.*
13. *QDyLoRA: Quantized Dynamic Low-Rank Adaptation for Efficient Large Language Model Tuning Hossein Rajabzadeh12, Mojtaba Valipour 1, Tianshu Zhu 2, Marzieh Tahaei arXiv:2402.10462v1 [cs.LG] 16 Feb 2024*
14. *Llama 2: Open Foundation and Fine-Tuned Chat Models. Hugo Touvron, Louis Martin, Kevin Stone and et al. arXiv:2307.09288v2 [cs.CL] 19 Jul 2023.*
15. *Exploring Parameter-Efficient Fine-Tuning Techniques for Code Generation with Large Language Models M. Weyssow, Xin Zhou, K. Kim et al. arXiv:2308.10462v2 [cs.SE] 18 Jan 2024.*
16. *CodePori: Large Scale Model for Autonomous Software Development by Using Multi-Agents. Zeeshan Rasheed, Muhammad Waseem, Mika Saari, Pekka Abrahamsson et al. arXiv:2402.01411v1 [cs.SE] 2 Feb 2024.*
17. *CRUXEval: A Benchmark for Code Reasoning, Understanding and Execution. Alex Gu, Baptiste Roziere, Hugh Leather et al. arXiv:2401.03065v1 [cs.SE] 5 Jan 2024.*
18. *MultiPL-E: A Scalable and Polyglot Approach to Benchmarking Neural Code Generation. Federico Cassano, John Gouwar, Daniel Nguyen et al. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 49, NO. 7, JULY 2023.*
19. *OOP: Object-Oriented Programming Evaluation Benchmark for Large Language Model. Shuai Wang, Liang Ding, Li Shen et al. arXiv:2401.06628v2 [cs.CL] 21 Feb 2024.*
20. *Mixtral of Experts. Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux. et al. arXiv:2401.04088v1 [cs.LG] 8 Jan 2024.*

МЕТОД ОБНАРУЖЕНИЯ ФАКТОВ ОБХОДА БЛОКИРОВОК РЕСУРСОВ СЕТИ ИНТЕРНЕТ

Ишкуватов С. М.[1], Бегаев А. Н.[2], Комаров И. И.[3], Левко И. В.[4]

DOI: 10.21681/2311-3456-2024-3-76-84

Цель исследования: *разработка и экспериментальное исследование метода обнаружения фактов обхода блокировки трафика, осуществляющего доступ к запрещённым Интернет-ресурсам.*

Методы исследования: *системный анализ, теория метрических пространств, математическая статистика, теория систем искусственного интеллекта, теория обработки экспериментальных данных.*

Полученные результаты: *систематизированы информативные признаки, используемые актуальными методами и средствами блокировки запрещённых ресурсов сети Интернет, а также способы обхода таких блокировок; определена новая совокупность информативных признаков, обеспечивающая решение задачи исследования; предложен обобщённый метод обнаруженя фактов обхода блокировки запрещённых ресурсов сети Интернет и получено экспериментальное подтверждение его продуктивности.*

Научная новизна *полученных результатов определяется систематизацией нормативно-правовых и организационно-технических требований к средствам обнаружения и блокирования доступа к запрещённым ресурсам сети Интернет, что обеспечивает формирование прогнозов их развития; использованием авторской совокупности методов мониторинга трафика на основании анализа цифровых отпечатков коммуникационных протоколов и закономерностей следования и объёма передаваемых данных, обеспечивающих возможность выявления и анализа информативных признаков обычно скрытых для пассивного наблюдателя; разработкой обобщённого метода обнаружения факта обхода блокировки трафика на основании анализа устойчивых закономерностей, присущих коммуникационным сессиям.*

Вклад авторов: *Бегаев А. Н. – определение технико-экономических ограничений и требований к реализации метода обнаружения факта обхода блокировки трафика; Комаров И. И. – постановка задачи и определение плана исследования; Ишкуватов С. М. – анализ информативных признаков, разработка метода обнаружения факта обхода блокировки трафика, проведение эксперимента; Левко И. В. – анализ нормативно-правовых аспектов регулирования доступа к Интернет-ресурсам, анализ и интерпретация результатов эксперимента.*

Ключевые слова: *Интернет-цензура, фильтрация трафика, туннелирование трафика, маскирование сессии, пассивный наблюдатель, цифровой отпечаток, глубокий анализ пакетов.*

A METHOD FOR DETECTING FACTS OF CIRCUMVENTION OF INTERNET RESOURCE LOCKS

Ishkuvatov S. M.[5], Begaev A. N.[6], Komarov I. I.[7], Levko I. V.[8]

The purpose of the study: *development and experimental study of a method for identifying facts of circumvention of traffic blocking, providing access to prohibited Internet resources.*

Research methods: *system analysis, theory of metric spaces, mathematical statistics, theory of artificial intelligence systems, theory of experimental data processing.*

1 Ишкуватов Сергей Маратович, аспирант факультета безопасности информационных технологий, Университет ИТМО, Санкт-Петербург, Россия. E-mail: sysroot0@gmail.com, ORCID ID: 0000-0002-4006-3693

2 Бегаев Алексей Николаевич, кандидат технических наук, генеральный директор АО «Эшелон-СЗ», Санкт-Петербург, Россия. E-mail: begaev@mail.ru, ORCID ID: 0000-0003-1186-7614

3 Комаров Игорь Иванович, кандидат физико-математических наук, доцент, доцент факультета безопасности информационных технологий, Университет ИТМО, Санкт-Петербург, Россия. E-mail: i_krov@mail.ru, ORCID ID: 0000-0002-6542-4950

4 Левко Игорь Владимирович, кандидат технических наук, доцент, Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, Россия. E-mail: levko_iv@mail.ru

5 Sergei M. Ishkuvatov, Ph.D. student, Faculty of Information Technology Security, ITMO University, St. Petersburg, Russia. E-mail: sysroot0@gmail.com

6 Alexey N. Begaev, Ph.D., CEO of JSC North-West Echelon, St. Petersburg, Russia. E-mail: begaev@mail.ru

7 Igor I. Komarov, Ph.D., (in Maht.), Associate Professor, Faculty of Information Technology Security, ITMO University, St. Petersburg, Russia. E-mail: i_krov@mail.ru

8 Igor V. Levko, Ph.D., Associate Professor, Mozhaisky Military Aerospace Academy, St. Petersburg, Russia. E-mail: levko_iv@mail.ru

The results obtained: *the informative signs used by current methods and means of blocking prohibited Internet resources, as well as ways to circumvent such locks, are systematized; a new set of informative signs providing a solution to the research problem is determined; a generalized method for detecting facts of circumventing the blocking of prohibited Internet resources is proposed and experimental confirmation of its productivity is obtained.*

The scientific novelty *of the results obtained is determined by the systematization of regulatory and organizational and technical requirements for means of detecting and blocking access to prohibited Internet resources, which ensures the formation of forecasts for their development; using the author's set of traffic monitoring methods based on the analysis of digital fingerprints of communication protocols and patterns of sequence and volume of transmitted data, providing the possibility of identifying and analyzing informative signs usually hidden to a passive observer; the development of a generalized method for detecting the fact of bypassing traffic blocking based on the analysis of stable patterns inherent in communication sessions.*

Contribution of the authors: *Begaev A. N. – definition of technical and economic limitations and requirements for the implementation of the method of detecting the fact of bypassing traffic blocking; Komarov I. I. – setting the task and defining the research plan; Ishkuvatov S. M. – analysis of informative signs, development of a method for detecting the fact of bypassing traffic blocking, conducting an experiment; Levko I. V. – analysis of regulatory aspects of regulating access to Internet resources, analysis and interpretation of experimental results.*

Keywords: *Internet censorship, traffic filtering, traffic tunneling, session masking, passive observer, digital fingerprint, deep packet analysis.*

Введение

Обеспечение информационной безопасности государства в условиях информационного противоборства сопряжено с разрешением объективного противоречия между соблюдением прав и свобод субъектов и необходимостью регулирования информационного потока в условиях глобальной доступности данных. С точки зрения технологических задач кибербезопасности выделяются ряд взаимосвязанных направлений, связанных с: выявлением и анализом сематического воздействия на пользователя [1, 2]; разработкой методов и средств анализа киберустойчивости сложных технических систем [3, 4]; совершенствованием методов реализации организационных решений в технических системах [5–7].

В контексте общего тренда развития правового обеспечения информационной безопасности России [8, 9], и в частности – согласно поправкам в Законы «О связи[9]» и «Об информации, информационных технологиях и защите информации[10]», вступившим в силу с 1 ноября 2019 года, операторы связи обязаны устанавливать специализированное оборудование для обеспечения безопасности и контроля передаваемой информации, в том числе – оборудование анализа и фильтрации трафика для ограничения доступа к запрещённым ресурсам сети Интернет, определённое в Законе как Технические Средства Противодействия Угрозам (ТСПУ). Закон предусматривает административную ответственность за нарушение требований по пропуску трафика через ТСПУ, а также уголовную ответственность за нарушение порядка их установки, эксплуатации и модернизации.

Одной из сложнейших задач практической реализации мер государственной политики в области кибербезопасности является обнаружение и управление трафиком, взаимодействующим с запрещёнными ресурсами. Эта задача осложняется использованием методов сокрытия самого факта обращения, высокой ресурсоёмкостью методов глубокого анализа трафика (DPI – Deep Packet Inspection), а также недостаточным уровнем развития научно-методического аппарата обнаружения такого трафика в общем потоке легитимных обращений, что приводит к низкой селективности используемых технических решений.

Таким образом актуализируется задача совершенствования научно-методического аппарата обнаружения и блокировки нежелательного трафика, особенно в условиях сознательного обхода запретов и ограниченности доступных вычислительных ресурсов.

Методы и средства блокировки нежелательного трафика

ТСПУ имеют целью фильтрацию трафика и блокировку доступа к запрещённым ресурсам сети Интернет. В Российской Федерации используются достаточно широкий спектр отечественных решений

9 Федеральный закон «Об информации, информационных технологиях и о защите информации» от 27.07.2006 N 149-ФЗ (в действующей редакции).

10 Федеральный закон «О связи» от 07.07.2003 N 126-ФЗ (в действующей редакции).

DPI[11], которые адаптированы для работы с единым реестром ресурсов Роскомнадзора[12]:

- ✓ СПАК «Equila» от ООО «Напа Лабс»,
- ✓ СПО «CyberFilter» от ИП Кучебо Н.Н.,
- ✓ СПО «Барьер» от АО «Энвижн Груп»,
- ✓ СПО «АДМ Filter» от ООО «АДМ Системы»,
- ✓ СПО «ZapretService» от ИП Пономаренко И.Р.,
- ✓ СПО «Ideco Selecta ISP» от ООО «Айдеко»,
- ✓ СПО «Carbon Reductor DPI» от ООО «Карбон Софт»,
- ✓ СПО «SkyDNS Zapret ISP» от ООО «СкайДНС»,
- ✓ СПАК «Тиксен-Блокировка» от ООО «Эд-АйТи»,
- ✓ СКАТ DPI от ООО «ВАС Экспертс»,
- ✓ СПАК EcoFilter от ООО «РДП.РУ»,
- ✓ СПО «UBIC» от ООО «Безопасный интернет»,
- ✓ САПК «Периметр-Ф» от ООО «МФИ Софт».

В зависимости от способа установки оборудования [10] возможны следующие типы блокировки.

- Пассивная блокировка – не предполагает работы в разрыв соединения и запрет обмена данными между узлами. При обнаружении признака запрещённого ресурса, в канал инжектируются пакеты завершения соединения. При такой организации оборудование DPI получает для анализа «отзеркалированный» трафик, а непосредственного запрета обмена не происходит.
- Активная блокировка, предполагающая работу в разрыв соединения и полноценную MITM-инжекцию в сессиях, имеющих признаки обращения к запрещённым ресурсам. Активная блокировка – ресурсозатратный подход: при превышении допустимой нагрузки могут возникнуть проблемы при передаче разрешённых сессий, поэтому обязательно применение механизма Bypass[13], который в случае перегрузки пустит трафик по альтернативному маршруту.

На практике в среде специалистов разрабатываются программные решения, позволяющие определить применяемые типы блокировок, например blockcheck[14].

Технически блокировка отдельной Web-страницы возможна лишь в случае использования протокола HTTP, доля которого непрерывно сокращается. Для протоколов, использующих шифрование HTTPS или QUIC, возможна только полная блокировка сессии. Согласно публичной документации приведённых выше ТСПУ можно сделать выводы о номенклатуре и эффективности применения информативных признаков, используемых для принятия решения о блокировке трафика, а именно:

- блокировка по IP-адресу – характеризуется простотой реализации, высокой производительностью, но крайне низкой селективностью: в случае использования сети доставки контента (CDN – Content Delivery Network) одновременно с запрещённым ресурсом будет заблокировано множество легитимных, поскольку одному IP-адресу соответствует множество сторонних ресурсов;
- блокировки по значениям полей HTTP HOST, HTTP URL являются достаточно универсальными, поскольку могут блокироваться только определённые Web-страницы. Однако протокол HTTP уже практически полностью вытеснен протоколом HTTPS, который не позволяет пассивному наблюдателю определить запрашиваемую страницу;
- блокировка по TLS SNI (Server Name Indication) не всегда применима, так как поле является опциональным, оно сообщает серверу к какому ресурсу обращается клиент во время TLS-рукопожатия;
- сертификат сервера, к которому обращается клиент: проверяется в том случае если не использовалось расширение TLS SNI.

Очевидно, что существующая практика блокировки запрещённых Интернет-ресурсов не в полной мере соответствует организационным решениям и правовым требованиям, что актуализирует задачу разработки методов автоматического обнаружения нежелательного трафика, обладающих повышенной селективностью и приемлемой ресурсоёмкостью, базирующихся на использовании новых информативных признаков.

Методы и средства обхода блокировки трафика

Блокировки Интернет-ресурсов, особенно случайные блокировки легитимных ресурсов, находящихся на одних серверах с запрещёнными, побуждают пользователей применять различные инструменты для обхода таких запретов. Известны следующие типы программных средств, применяемых для обхода ограничений и скрывающих от пассивного наблюдателя информативные признаки, по которым принимается решение о блокировке.

- Программы-фрагментаторы сессий, работа которых основана на том, что протокол TCP допускает нарушение хронологии передачи пакетов, их фрагментацию или потерю. Принимающая сторона, может восстановить исходную хронологию и перезапросить потерянные фрагменты. Известны следующие виды искусственной фрагментации:

11 Российские производители DPI и их платформы URL: https://vasexperts.ru/blog/dpi/rossijskie-proizvoditeli-dpi-i-ih-platformy/ (дата обращения: 10.02.2024).

12 Единый реестр доменных имён, указателей страниц сайтов в сети «Интернет» и сетевых адресов, позволяющих идентифицировать сайты в сети «Интернет», содержащие информацию, распространение которой в Российской Федерации запрещено -URL: https://eais.rkn.gov.ru/ (дата обращения: 10.02.2024)

13 Устройства Bypass предполагают коммутацию входного пакета на выход, минуя вышедшие из строя определённые функциональные блоки URL: https://moxa.pro/blogs/articles/obzor-bypass-ustroystva-obu-102 (дата обращения: 10.02.2024)

14 https://github.com/ValdikSS/blockcheck

- TCP-фрагментация для первого пакета данных;
- TCP-фрагментация пакетов, содержащих Keep-Alive;
- синтаксическое смешивание с целью обхода встроенных шаблонов ТСПУ, но с сохранением корректности с точки зрения спецификации протокола HTTP (произвольное изменение регистра букв; изменение пробельных символов; добавление пробелов к заголовку; перенос строк в Unix-стиле);
- введение в заблуждение DPI (отправка ложных пакетов с низким TTL, некорректными контрольными суммами, некорректным порядком следования TCP Sequence/Acknowledgment);
- фрагментация поля TLS Client Hello таким образом что часть имени сервера будет находиться в одном пакете, а продолжение в другом.

Задача детектирования таких сессий осложняется тем, что некоторые протоколы, такие как Jabber, могут начать процедуру TLS-рукопожатия после обмена нешифрованной служебной информацией, что означает, что дефрагментация и анализ пакета не может ограничиваться только первыми несколькими пакетами сессии. Искусственная фрагментация пакетов сессии позволяет разбить передаваемый признак на разные пакеты[15], тем самым сделать невозможным его определение без полной фрагментации и сборки сессии. Поскольку все приведённые выше признаки, кроме IP-адреса, не всегда передаются в первом пакете сессии, оборудование ТСПУ должно либо резервировать вычислительные ресурсы и память для дефрагментации каждой проходящей через него сессии, либо выявлять признаки блокировки только в целых пакетах.

- Программы, использующие eSNI или Encrypted Client Hello и позволяющие скрыть доменное имя запрашиваемого ресурса: предполагается одновременное использование протоколов DNS over TLS, DNS over HTTPS, DNS over QUICK или других протоколов, шифрующих запросы DNS.
- Программы, использующие трудно детектируемые протоколы (например, Telegram и некоторые протоколы BitTorrent).
- Использование туннелирования трафика различными VPN-решениями: предполагается наличие сервера за оборудованием ТСПУ, соединение с которым осуществляется с помощью туннеля (например, OpenVPN, IPsec, Wireguard). В таком случае все приведённые информативные признаки передаются в шифрованном виде, исключающем их выявление оборудованием ТСПУ.
- Туннелирование трафика инструментами[16], не являющимися распространёнными VPN-решениями (например, туннелирование TLS over SSH, TLS over TLS с использованием программ Shadowsocks, OCserv).
- Туннелирование с использованием стеганографии – организуется туннель между абонентом и сервером, находящемся за ТСПУ. Однако скрывается сам факт использования туннеля: трафик маскируется под другой тип (например, инструмент XRay, который маскирует трафик под TLS-сессии популярных приложений, воспроизводя их цифровые отпечатки (ЦО)).

Таким образом современное состояние противоборства технологий блокировки трафика и их обхода характеризуется следующими тезисами.

- ✓ С увеличением доступных вычислительных ресурсов методы, связанные с фрагментацией пакетов, теряют актуальность: оборудование ТСПУ дефрагментирует сессии или их начальные пакеты до завершения процедуры рукопожатия сторон.
- ✓ Все сессии, использующие расширение eSNI, могут быть заблокированы при обнаружении поддержки такого расширения клиентом в процессе TLS-рукопожатия.
- ✓ Протоколы, шифрующие DNS-запросы, могут быть заблокированы по конечным точкам; их блокировка по IP-адресу не должна повлиять на доступность других ресурсов.
- ✓ Очевидно, что блокировки только нешифрованных ответов DNS, содержащих адреса запрещённых ресурсов, недостаточно ввиду массового распространения альтернативных протоколов, использующих шифрование.
- ✓ Блокировка всех протоколов, которые не удалось идентифицировать ТСПУ, приведёт к блокировкам множества частных нераспространённых протоколов и неработоспособности множества простых сетевых устройств и не распространённых сервисов.
- ✓ Блокировка всех VPN-соединений, также невозможна, такт как этот протокол легитимно используется множеством организаций для обеспечения связи своих филиалов, удалённого доступа сотрудников во внутреннюю сеть или личные рабочие места через сеть Интернет.
- ✓ Допустимой является блокировка сессий с конечными точками известных VPN-сервисов анонимайзеров, но такой подход будет иметь ограниченную эффективность виду большого числа подобных сервисов и относительной простоты их миграции.

15 Автономный способ обхода DPI и эффективный способ обхода блокировок сайтов по IP-адресу. URL: https://habr.com/ru/post/335436 (дата обращения: 10.02.2024)

16 Современные технологии обхода блокировок: V2Ray, XRay, XTLS, Hysteria, Cloak и все-все-все. URL: https://habr.com/ru/articles/727868/ (дата обращения: 10.02.2024).

✓ Сохраняется основное противоречие организации блокировки в условиях плохой селективности: при строгой блокировке всегда будут затронуты сторонние ресурсы и сервисы, а мягкая не обеспечивает достижения поставленных целей.

Постанова задачи, гипотеза исследования и эксперимент

Задача исследования состоит в разработке метода обнаружения обходов блокировок трафика, базирующегося на новых информативных признаках и позволяющего отличить штатное использование протоколов от их применения в качестве инструментария обхода блокировок.

Гипотеза исследования: «*Средства, используемые для обхода блокировок трафика, обладают устойчивыми информативными признаками, сохраняющимися при применении стандартных методов их использования*».

Проверка гипотезы исследования проведена экспериментальным путём с применением авторского теоретического аппарата [11–13], расширяющего возможности принятия решения в задачах кибербезопасности за счёт анализа ЦО коммуникационных протоколов и выявленных закономерностей между порядком следования и объёмом передаваемых данных в процессе взаимодействия.

Экспериментальный стенд включает компьютер с ОС Windows и облачный Linux-сервер, между которыми организовывались туннели с помощью широко распространённых программ OCserv и XRay.

Для получения записей на сервере использованы средства screen, tcpdump и сервер телефонии Asterisk. На клиентской части – nekoray[17] и VoIP-клиент. Записаны сессии инструмента TLS over TLS OCserv и сессии туннелей XRay для трафика Web-браузера и тестового VoIP-звонка.

Для оценки качества детектирования сохранены обычные TLS-сессии, не являющиеся сессиями инструментов туннелирования. С помощью авторского инструментария все сессии дефрагментировались и исследовались с целью выявления устойчивых закономерностей.

Информативные признаки сессий туннелей, организованных OCserv

Демаскирующие признаки туннелирования трафика инструментами, не являющимися распространёнными VPN-протоколами, определяются тем, что средство, которое реализует туннель TLS over TLS или SSH over TLS создаёт сессии, выделяющиеся *продолжительностью*, а также частичным сохранением *объёмных закономерностей* исходного (маскируемого) трафика. Пассивному наблюдателю доступны признаки внешнего TLS или SSH-рукопожатия, а такие рукопожатия, в свою очередь, также имеют свои ЦО JA3(JA4), HASSH.

Экспериментально подтверждено, что все сессии инструмента OCserv, несмотря на возможность маскировки под обычные HTTPS-сессии за счёт задания произвольных конечных точек TLS Server Name, ЦО TLS соответствует реализации OpenConnect, которая не используется иначе, как для организации туннелей, что является явным признаком попытки маскирования туннелированной сессии.

Наблюдаемый ЦО отличается от ЦО популярных браузеров и также может однозначно характеризовать сессию туннеля. В табл. 1 приведены известные значения ЦО, полученные из базы данных Cisco Mercury[18], полужирным шрифтом выделена строка с ЦО, соответствующая сессиям туннелей.

Естественно, что признаком для блокировки трафика, генерируемого данным средством, будет обнаружение такого ЦО TLS, кроме того, должен быть заблокирован трафик и его новых реализаций, содержащий ЦО близких соседей, найденных по методу [11].

Дополнительными демаскирующими признаками таких сессий являются:

– распределения длин пакетов, сильно отличающихся от остальных сессий, не являющимися туннелями;
– аномальная частота появления сессий с подобным адресатом.

Информативные признаки стеганографически туннелированного трафика

Типичным примером программы для стеганографического сокрытия туннелированного трафика является XRay. Будучи установлен на сервер, он переадресует все HTTPS-запросы не от своей клиентской части на запрашиваемый ресурс. Клиентская часть в точности воспроизводит процесс TLS-рукопожатия с произвольно выбранным ЦО; проблемы поддержки всех возможных опций и алгоритмов шифрования нет, поскольку серверная часть в любом случае проигнорирует их и ответит сообщением TLS-рукопожатия сервера с постоянным ЦО JA3S, после чего начнётся обмен шифрованными пакетами.

Пакеты TLS-рукопожатия отправляются сторонами исключительно с целью ввести в заблуждение пассивного наблюдателя и убедить его в том, что сессия является обычной сессией TLS. Выявление возможно по заранее известным последовательностям обмена информацией.

Несмотря на то, что ЦО TLS-сессии туннеля может быть задан произвольно, все они не являются сессиями TLS, а только выглядят так для пассивного

17 Ресурс разработки nekoray URL: https://github.com/MatsuriDayo/nekoray (дата обращения: 10.02.2024).

18 База данных ЦО Cisco Mercury URL: https://github.com/cisco/mercury/blob/main/resources/fingerprint-db-tls-os.json.gz (дата обращения: 10.02.2024)..

Таблица 1.

Известные ЦО различных версий OpenConnect

JA3 полное представление	JA4
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49187-49200-52392-49172-49199-49171-49191-157-49309-53-61-156-49308-47-60-159-52394-49311-57-107-158-49310-51-103,5-10-11-13-35-51-43-65281-0-45-28-21,23-24-25-29-256-257-258-259-260,0	t13d351200_bfa337485184_b4c318310b83
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49187-49200-52392-49172-49199-49171-49191-157-49309-53-61-156-49308-47-60-159-52394-49311-57-107-158-49310-51-103,5-10-11-13-35-51-43-65281-0-45-28-21,23-24-25-29-30-256-257-258-259-260,0	**t13d351200_bfa337485184_5671b5df5029**
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49187-49200-52392-49172-49199-49171-49191-157-49309-53-61-156-49308-47-60-159-52394-49311-57-107-158-49310-51-103,5-10-11-13-35-51-43-65281-45-28-21,23-24-25-29-256-257-258-259-260,0	t13i351100_bfa337485184_b4c318310b83
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49187-49200-52392-49172-49199-49171-49191-157-49309-53-61-156-49308-47-60-159-52394-49311-57-107-158-49310-51-103,5-10-11-13-35-51-43-65281-45-28-21,23-24-25-29-30-256-257-258-259-260,0	t13i351100_bfa337485184_5671b5df5029
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49200-52392-49172-49199-49171-157-49309-53-156-49308-47-159-52394-49311-57-158-49310--51,5-10-11-13-35-51-43-65281-45-28-21,23-24-25-29-256-257-258-259-260,0	t13i291100_723694b0fccc_b4c318310b83
771,4866-4867-4865-4868-49196-52393-49325-49162-49195-49324-49161-49200-52392-49172-49199-49171-157-49309-53-156-49308-47-159-52394-49311-57-158-49310--51,5-10-11-13-35-51-43-65281-45-28-21,23-24-25-29-256-257-258-259-260,0	t13i291100_723694b0fccc_b4c318310b83
771,49195-49196-49286-49287-49161-49187-49162-49188-49266-49267-49324-49325-49160-49199-49200-49290-49291-49171-49191-49172-49192-49270-49271-49170-156-157-49274-49275-47-60-53-61-65-186-132-192-49308-49309-10-158-159-49276-49277-51-103-57-107-69-190-136-196-49310-49311-22,5-65281-35-10-11-13,23-24-25-21-19,0	t12i540600_a499d9840d02_10551b21ac36
771,49196-49287-52393-49325-49162-49188-49267-49195-49286-49324-49161-49187-49266-49160-49200-49291-52392-49172-49192-49271-49199-49290-49171-49191-49270-49170-157-49275-49309-53-61-132-192-156-49274-49308-47-60-65-186-10-159-49277-52394-49311-57-107-136-196-158-49276-49310-51-103-69-190-22,5-65281-35-10-11-13,23-24-25,0	t12i570600_45f33a1adcc2_10551b21ac36

наблюдателя. Протокол TLS разделяет данные, отправляемые сторонами, на субпакеты; для каждого субпакета указываются его тип и длина, которые доступны пассивному наблюдателю. Соответственно, он может отслеживать процесс установления соединения и хронологию обмена данными сторонами, анализируя типы субпакетов и их объёмы.

Субпакеты типа CLIENT_HELLO имеют ЦО, который может быть произвольно выбран из обширного списка. Каждый клиентский пакет TLS-рукопожатия всегда содержит поле идентификатор сессии (Session ID) с целью убедить пассивного наблюдателя, что сессия является возобновлением некой предыдущей сессии, чтобы сократить процедуру рукопожатия и легитимировать отсутствие в ответе сервера его сертификата.

Субпакеты туннеля SERVER_HELLO всегда имеют ЦО JA3S «15af977ce25de452b96affa2addb1036», который не меняются из сессии в сессию.

Субпакеты типа APPLICATION_DATA, содержат шифрованную информацию, которая не может быть дешифрована пассивным наблюдателем без наличия

сессионных ключей. Однако длина этих полей известна из заголовка, она определяется размером зашифрованного сообщения и алгоритмом шифрования, выбранным сервером.

Возможность выделять типы и размеры пакетов позволяет пассивному наблюдателю синтезировать сценарии взаимодействия и отслеживать объёмы передаваемых данных. Все сессии исследованных туннелей начинались одинаково по сценарию (Листинг 1).

Листинг 1.

Начало сессий туннелей, организованных XRay

```
CLIENT              SERVER
C:CLIENT_HELLO,
                    S:SERVER_HELLO,
                    S:CHANGE_CIPHER_SPEC,
                    S:APPLICATION_DATA(52),
                    S:APPLICATION_DATA(5562),
                    S:APPLICATION_DATA(281),
                    S:APPLICATION_DATA(69),
C:CHANGE_CIPHER_SPEC,
C:APPLICATION_DATA(69),
…
```

где: C – пакет от клиента, S – пакет от сервера, в скобках – постоянные размеры пакетов.

При туннелировании VoIP-сессии равномерно передаваемые RTP-пакеты кодека G.711, имеющие полную длину 200 байт, порождают в сессии туннеля пакеты APPLICATION_DATA длиной 205 байт, с периодичностью ≈ 0,02 сек. Например, симплексная сессия туннеля XRay (рис. 1), замаскированная под сессию TLS в момент VoIP звонка, использующего RTP-кодек G.711 PCMU [14].

Подобная закономерность позволяет выявлять VoIP-сессии в туннеле с помощью искусственных нейронных сетей [15–17] или без них, как это предложено в работе [18].

Метод обнаружения фактов обхода блокировок ресурсов сети Интернет

Обобщённый метод обнаружения фактов обхода блокировок ресурсов сети Интернет способом скрытого туннелирования трафика, использованный в работе, предполагает последовательный анализ доступных для пассивного наблюдателя устойчивых информативных признаков, причём сложность последующих этапов возрастает.

В первую очередь выполняется вычислительно простая:

- проверка ЦО TLS на строгое соответствие ЦО реализации OpenConnect,
- в случае получения неизвестного ЦО выполняется его проверка на удалённость от ЦО OpenConnect по методу [11].

Для выявления сессий, использующих стеганографию:

- беспрепятственно пропускаются все первичные сессии, использующие полную процедуру рукопожатия, которая не применяется инструментами обхода блокировок ресурсов;
- проверяются ЦО ответа сервера JA3S;
- при отсутствии признаков нарушений на предыдущих этапах – проверяется соответствие сценария обмена данными после рукопожатия.

Среди оставшихся сессий следует выбрать количественно доминирующие сессии с конечными точками за границей ТСПУ.

Только для оставшихся и соответствующих всем критериям сессий проводить анализ распределения длин и периодов следования пакетов.

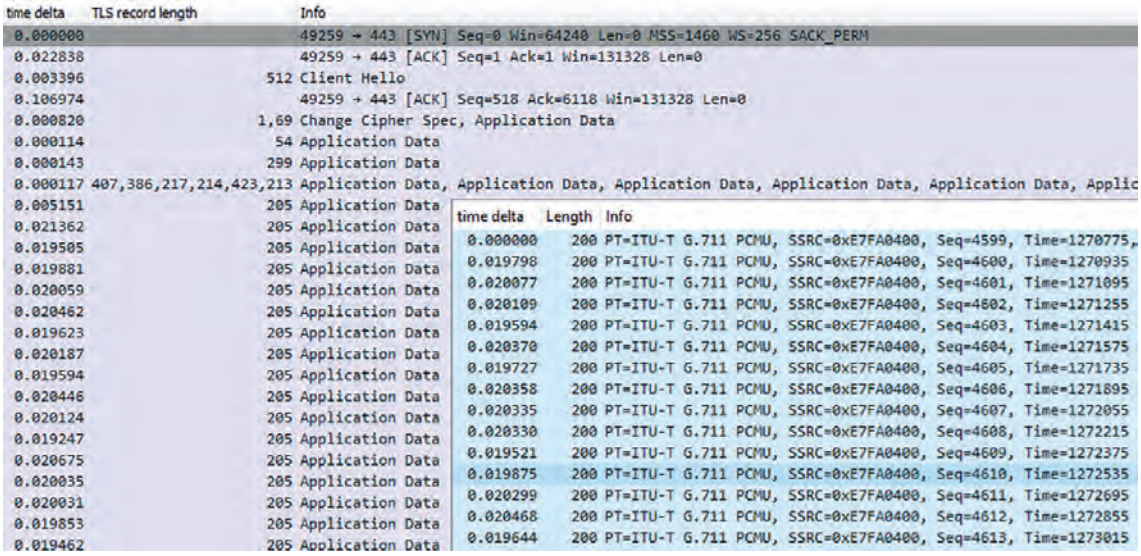


Рис. 1. Сессия туннеля XRay, замаскированная под сессию TLS (слева), нешифрованные пакеты RTP-сессии (справа)

По результатам экспериментальной проверки предлагаемого обобщённого метода получены следующие характеристики качества обнаружения скрытого туннелирования трафика:

- Все сессии, организованные средством OCserv, однозначно идентифицированы по ЦО TLS. Можно утверждать, что данный ЦО всегда будет соответствовать только сессиям туннелей и никогда – сессиями Web-браузера.
- Все 1975 исследованных сессий имели несовпадающие для разных сессий значения Session ID, строгое совпадение объемнохронологического сценария обмена субпакетами в начале каждой сессии.
- Для сессии XRay содержащей VoIP G.711 результат работы искусственной нейронной сети, спроектированной для исследования трафика OpenVPN:

FTP: 6.44%	(0.06437485)
RTP G.711: 76.7%	(0.7669719)
RTP G.723.1: 0.14%	(0.0014005811)
RTP G.729: 1.53%	(0.015301358)
HTTP: 0.02%	(0.00019485639)
RDP: 14.01%	(0.14014894)
SMB: 1.16%	(0.011607527)

Выводы

Поиск способов выявления сессий, являющихся скрытыми туннелями, а также новых способов их сокрытия идут параллельно. Обнаружение информативных признаков, однозначно идентифицирующих такие туннели, приводит к попыткам их сокрытия в новых версиях ПО. Вместе с тем, искусственное маскирование туннелированных сессий, в том числе для сокрытия от пассивного наблюдателя вновь обнаруживаемых информативных признаков (устойчивых закономерностей), приводят к снижению эффективности коммуникации.

В результате исследования выдвинута и экспериментально подтверждена гипотеза о сохранении доступности пассивному наблюдателю устойчивых информативных признаков стандартного функционирования средств, используемые для обхода блокировок трафика.

В частности, показаны примеры успешного детектирования туннелей при передаче по ним не типичных видов трафика (например VoIP). В данном случае использованы сохраняющиеся закономерности распределения длин пакетов с поправкой на объём служебной информации.

В качестве примера детектирования стеганографически организованного туннеля проведён анализ сессий, организованных с помощью инструмента XRay. В результате обнаружена устойчивая хронологическая последовательность сообщений, что позволяет безошибочно определять такие сессии.

С точки зрения оптимизации вычислительных затрат на анализ трафика предложена последовательность шагов обобщённого метода обнаружения фактов обхода блокировок ресурсов сети Интернет: последовательно усложняющиеся процедуры проверки обеспечивают потапный контроль правил и снижение числа сессий, требующих обработки с использованием систем искусственного интеллекта. Ей должны подвергаться только сессии, имеющие все косвенные признаки: присутствие Session ID, совпадение известный ЦО ответа сервера JA3S, совпадение последовательности обмена при установлении соединения.

Литература

1. *Чеповский А. А. Об особенностях построения и анализа графов взаимодействующих объектов в сети telegram-каналов // Вопросы кибербезопасности. – 2022. – №. 1 (53), с. 75–81. DOI:10.21681/2311-3456-2022-2-75-81*
2. *Капицын С. Ю., Рюмшин К. Ю., Варенница В. В. Логико-лингвистический механизм формирования «бумажных» пуль при информационном противоборстве // Вопросы кибербезопасности. – 2022. №. 1 (53), с. 93–99. DOI:10.21681/2311-3456-2022-1-93-99*
3. *Новикова Е. С. и др. Обнаружение вторжений на основе федеративного обучения: архитектура системы и эксперименты // Вопросы кибербезопасности. – 2023. – № 6 (58), с. 50–66. DOI:10.21681/2311-3456-2023-6-50-66*
4. *Коноваленко С. А. Методика оценивания информационной устойчивости гетерогенной системы обнаружения компьютерных атак // Вопросы кибербезопасности. – 2023. – №. 6 (58), с. 67–80. DOI:10.21681/2311-3456-2023-6-67-80*
5. *Шадрина А. Д. Способы защиты информации в веб-приложении // Программно-техническое обеспечение автоматизированных систем. – 2021. – с. 116–119.*
6. *Гурина Л. А., Айзенберг Н. И. Поиск эффективного решения по обеспечению защиты от киберугроз сообщества микросетей со взаимосвязанными информационными системами // Вопросы кибербезопасности. – 2023. – №. 3 (55). – с. 37–49. DOI:10.21681/2311-3456-2023-3-37-49*
7. *Павленко Е. Ю. и др. Распознавание киберугроз на адаптивную сетевую топологию крупномасштабных систем на основе рекуррентной нейронной сети // Вопросы кибербезопасности. – 2022. – №. 6 (52), с. 93–99. z DOI:10.21681/2311-3456-2022-6-93-99*
8. *Добродеев А. Ю. Кибербезопасность в Российской Федерации. Модный термин или приоритетное технологическое направление обеспечения национальной и международной безопасности XXI века // Вопросы кибербезопасности. – 2021. – №. 4 (44). – С. 61–72. DOI:10.21681/2311-3456-2021-4-61-72*
9. *Карцхия А. А. Новые элементы национальной безопасности: национальный и международный аспект // Вопросы кибербезопасности. – 2020. – №. 6 (40). – С. 72–82. DOI:10.21681/2311-3456-2020-6-72-82*
10. *VAS Experts. СКАТ – Система контроля и анализа трафика. VAS Experts. URL: https://vasexperts.ru/wp-content/uploads/2022/07/filtracziya-po-spiskam-rkn-i-minyusta.pdf (дата обращения: 10.02.2024)*

11. *Ишкуватов С. М., Швед В. Г., Филькова И. А. Метод оценки близости цифровых отпечатков реализаций протоколов // Защита информации. Инсайд. – №. 2. – с. 29–33.*
12. *Ишкуватов С. М., Комаров И. И. Анализ аутентичности трафика на основании данных цифровых отпечатков реализаций сетевых протоколов // Научно-технический вестник информационных технологий, механики и оптики. – 2020. – Т. 20. – №. 5. – С. 747–754.*
13. *Ишкуватов С. М., Бегаев А. Н., Комаров И. И. Метод автоматической классификации цифровых отпечатков TLS-протокола // Вопросы кибербезопасности. – 2024. – №. 1 (59), с. 67–74. DOI:10.21681/2311-3456-2024-1-67-74*
14. *Henning Schulzrinne, Stephen Casner, Ron Frederick, Van Jacobson. RTP: A transport protocol for real-time applications. RFC 3550. 2003 г.*
15. *Ali Rasteh, Florian Delpech, Carlos Aguilar-Melchor et al. Encrypted internet traffic classification using a supervised spiking neural network. Neurocomputing. 2022 г., Т. 503., 8.*
16. *Gupta Neha, Jindal Vinita, Bedi Punam. Encrypted traffic classification using extreme gradient boosting algorithm. International Conference on Innovative Computing and Communications. 2022 г., Т. Volume 3 / Springer., 9.*
17. *Islam Faiz Ul, Liu Guangjie, Liu Weiwei. Identifying VoIP traffic in VPN tunnel via flow spatio-temporal features. Mathematical Biosciences and Engineering. 2020 г., Т. 15, 5.*
18. *Ишкуватов, С. М. Способ и алгоритм определения типа трафика в шифрованном канале связи // Труды учебных заведений связи. 2022 г., Т. 8, 4.*

МЕТОД ОБНАРУЖЕНИЯ ПРОГРАММ-ВЫМОГАТЕЛЕЙ НА ОСНОВЕ АНАЛИЗА ПОВЕДЕНЧЕСКОГО ОТЧЕТА ИСПОЛНЯЕМОГО ОБЪЕКТА

Стародубов М. И.[1], Артемьева И. Л.[2], Селин Н. А.[3]

DOI: 10.21681/2311-3456-2024-3-85-89

Цель работы: *разработка метода обнаружения программ-вымогателей на основе анализа последовательностей API-вызовов и системных вызовов.*

Метод исследования: *анализ записей в поведенческом отчёте продукта виртуализации с использованием алгоритма глубокого обучения DeBERTa-V3.*

Полученный результат: *несмотря на большое разнообразие семейств и вариаций семейств программ-вымогателей, используемых злоумышленниками в компьютерных атаках, все они оставляют следы своей работы в атакуемой инфраструктуре. Одним из способов выявления вредоносного программного обеспечения и предотвращения заражения является использование технологии «Песочница», в том числе для выявления скрытых возможностей исследуемого объекта и аномалий его поведения. Функционирование любой компьютерной программы можно представить в виде набора записей его действий в отчёте поведения, которые можно рассматривать в качестве признаков объекта. В работе проведен анализ отчётов поведения программ-вымогателей. На сформированном наборе данных с использованием алгоритма глубокого обучения построена модель, позволяющая в дальнейшем выявлять вредоносные объекты, а также описан метод обнаружения программ-вымогателей.*

Практическая ценность *состоит в создании метода обнаружения программ-вымогателей на основе анализа поведенческого отчета исполняемого объекта с использованием алгоритма глубокого обучения DeBERTa-V3.*

Ключевые слова: *вредоносное программное обеспечение, песочница, глубокое обучение, BERT, Ransomware, компьютерные атаки.*

A METHOD FOR DETECTING RANSOMWARE BASED ON THE ANALYSIS OF THE BEHAVIORAL REPORT OF THE EXECUTABLE OBJECT

Starodubov M. I.[4], Artemyeva I. L.[5], Selin N. A.[6]

The aim of the work *is to develop a method for detecting ransomware based on the analysis of sequences of API calls and system calls.*

The research method *is the analysis of records in the behavioral report of the virtualization product using the deep learning algorithm DeBERTa-V3.*

The result obtained: *despite the wide variety of families and variations of the ransomware family used by attackers in computer attacks, they all leave traces of their work in the attacked infrastructure. One of the ways to identify malicious software and prevent infection is to use the Sandbox technology, including to identify the hidden capabilities of the object under study and anomalies of its behavior. The functioning*

1 Стародубов Максим Игоревич, аспирант ФГАОУ ВО «Дальневосточный федеральный университет» (ДВФУ), г. Владивосток, Россия. E-mail: starodubov.mi@dvfu.ru
2 Артемьева Ирина Леонидовна, доктор технических наук, профессор, заместитель директора по науке, профессор департамента программной инженерии и искусственного интеллекта Института математики и компьютерных технологий (Школы) ФГАОУ ВО «Дальневосточный федеральный университет» (ДВФУ), г. Владивосток, Россия. E-mail: artemeva.il@dvfu.ru
3 Селин Никита Александрович, студент департамента Информационной безопасности ФГАОУ ВО «Дальневосточный федеральный университет» (ДВФУ), г. Владивосток, Россия. E-mail: selin.na@dvfu.ru

4 Maxim I. Starodubov, Ph. D. student, Far Eastern Federal University (FEFU), Vladivostok, Russia. E mail: starodubov.mi@dvfu.ru
5 Irina L. Artemyeva, Dr. Sc., Professor, Deputy Director for Scientific Work at the Institute of Mathematics and Computer Technology (School) of the Far Eastern Federal University (FEFU), Vladivostok, Russia. E mail: artemeva.il@dvfu.ru
6 Nikita A. Selin, student of the Information Security Department of the Far Eastern Federal University (FEFU), Vladivostok, Russia. E-mail: selin.na@dvfu.ru

of any computer program can be represented as a set of records of its actions in a behavior report, which can be considered as signs of an object. The paper analyzes reports on the behavior of ransomware programs. Based on the generated data set using a deep learning algorithm, a model is built that allows further detection of malicious objects, and a method for detecting ransomware is described.

***The scientific novelty** consists in the creation of a method for detecting ransomware based on the analysis of the behavioral report of an executable object using the deep learning algorithm DeBERTa-V3.*

***Keywords:** malware, sandbox, deep learning, BERT, Ransomware, computer attacks.*

Введение

Проникновение компьютерных технологий в нашу жизнь приводит к её значительной от них зависимости и вызывает большой интерес злоумышленников. Доля атак[7] с использованием вредоносного программного обеспечения (ВПО, вредоносного ПО) на устройства пользователей превышает 55 %. Наибольший рост модификаций вредоносного ПО наблюдается в классе «программ-вымогателей», в 2023 году раз в 8 дней появлялось новое семейство, а каждые 22 минуты появлялась модификация известных семейств[8]. В связи с этим не теряет своей актуальности проблема обнаружения вредоносного программного обеспечения.

Существует два подхода к анализу программного обеспечения на наличие вредоносной составляющей. Статический анализ изучен хорошо и его эффективность достигает 99.4% [1]. Однако данный вид анализа неэффективен против сложных разновидностей вредоносных программ [2], которые используют методы шифрования [3], скрытия [4], упаковки [5] и полиморфных, олигоморфных и метаморфных преобразований [6]. Обнаружению при помощи динамического анализа уделено не так много внимания [7], хотя этот путь и выглядит более эффективным и более перспективным [8]. Данный вид анализа может помочь получить информацию о последовательности API-вызовов и системных вызовов, которые могут являться индикаторами возможного вредоносного поведения [9]. Однако такие последовательности могут быть очень длинными и сложными для понимания.

В связи с этим остро встаёт вопрос автоматизации их обработки. В контексте обнаружения ВПО для этого используются такие методы, как Word2Vec [10], HMM2Vec [11], BERT [12] и ELMo [13]. BERT показал наибольшую эффективность в системах MalBERT [14] и её усовершенствованной версии MalBERTv2 [15]. Однако, в указанных выше системах используется подход статического анализа. В литературе не было найдено исследований, показывающих связь указанных выше методов с подходом динамического анализа.

Метод обнаружения

Предлагаемый метод обнаружения ВПО основан на анализе и обработке поведенческого отчета исполняемого объекта. Поведенческий отчёт является одной из самых важных частей анализа вредоносных объектов, наряду с исходным кодом, и может дать полное представление о скрытых возможностях исследуемого объекта.

Пусть $Event = \{Event_1,...,Event_k\}$ – множество всех отслеживаемых программой событий. Размер данного множества зависит от средства виртуализации и среды выполнения объекта.

Основной алгоритм:

1. На основе поведенческого отчета исполняемого объекта «обработчик» формирует вектор доступных событий программы в среде выполнения $\tilde{U} = \{Event_1,...,Event_l\}$;
2. Сформированный вектор $\tilde{U}$ подается на вход обученного модуля DeBERTa-V3 [16]. На выходе имеется вектор значимых характеристик SC (significant characteristics);
3. Полученный вектор SC подается на вход заранее обученного классификатора, который на выходе выдает результат $D = \{0,1\}$.

Значение $D = 0$ соответствует тому, что исполняемый объект не является ВПО, а $D = 1$ – исполняемый объект является ВПО.

Схема предлагаемого метода обнаружения ВПО представлена на рисунке:

Рис. 1. Схема метода обнаружения ВПО

Для обучения элементов предлагаемого метода обнаружения ВПО требуется сформировать набор поведенческих отчетов исполняемых объектов из ряда ВПО и не ВПО.

7 Актуальные киберугрозы: IV квартал 2022 года [Электронный ресурс]. – Режим доступа: URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2022-q4/ (дата обращения: 02.01.2024).

8 Kaspersky Security Bulletin 2023. Statistics [Электронный ресурс]. – Режим доступа: URL: https://securelist.com/ksb-2023-statistics/111156/ (дата обращения: 02.01.2024).

Формирование набора отчетов поведения для обучения происходит следующим образом:

1. Исполняемый объект отправляется на анализ;
2. На физическом компьютере запускается виртуальная машина с известными параметрами и ей на исполнение передаётся анализируемый объект;
3. Измененные после исполнения объекта параметры передаются обратно на физический компьютер;
4. Формируется поведенческий отчёт объекта.

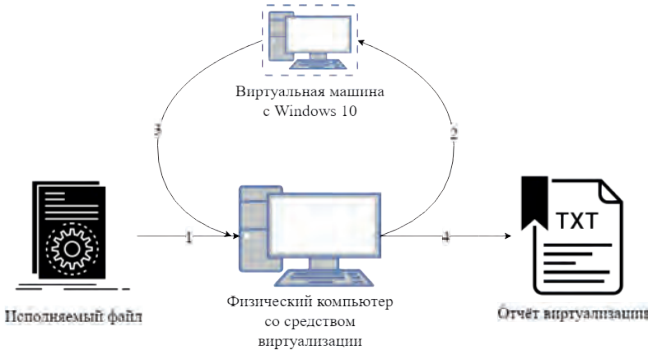


Рис. 2. Получение поведенческого отчёта

Набор отчетов поведения с использованием «обработчика» преобразовывается в массив доступных событий:

$U = \{U_1,...,U_n\}$, где $U_i = \{Event_{i,1},...,Event_{i,l}\}$, n – количество исполняемых объектов для обучения.

Затем с использования функционала $\varphi(U_i)$, обозначающего выполнение программы U_i и приводящий либо к безопасному состоянию системы «0», либо небезопасному состоянию «1», набор $\boldsymbol{U}$ преобразуется в вектор:

$$P = \{P_1,...,P_n\},$$

где $P_i = \varphi(U_i) = \{0,1\}$, отвечающий за принадлежность исполняемого объекта к ВПО.

В основе модуля DeBERTa-V3 лежит метод BERT [17] – метод обработки данных, основанный на трансформерах.

Из-за особенностей DeBERTa-V3 набор $\boldsymbol{U}$ разделяется на $\boldsymbol{B}$ и $\boldsymbol{R}$, где:

$$U_i \in \boldsymbol{B}, \text{ если } P_i == 1;$$
$$U_i \in \boldsymbol{R}, \text{ если } P_i == 0.$$

После обучения набор $\boldsymbol{U}$ обрабатывается DeBERTa-V3 и на выходе получаются значимые характеристики $\boldsymbol{SC}$.

Затем на основе $\boldsymbol{SC}$ и значений $\boldsymbol{P}$ производится обучение «классификатора».

Испытание предлагаемого метода

В этом разделе представлен эксперимент, который является испытанием предложенного метода. Общее представление всего эксперимента на высоком уровне представлено на рисунке 3.

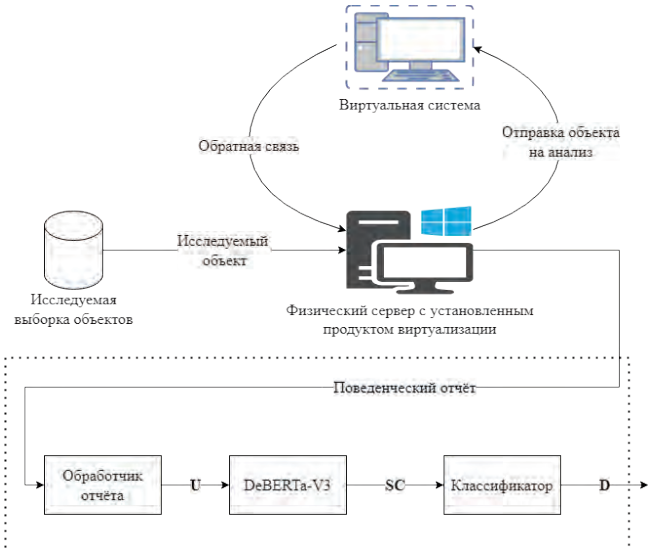


Рис. 3. Схема эксперимента

Процесс состоит из трех основных этапов:

- ✓ создание набора данных;
- ✓ этап предварительной обработки;
- ✓ этап точной настройки.

Набор данных

В качестве исследуемого множества была сформирована выборка их 600 000 файлов, обладающих следующими свойствами:

1. Все файлы являются исполняемыми объектами в системе ОС Windows и имеют формат PE[9];
2. Файлы должны эмулироваться средствами Cuckoo Sandbox;
3. Вредоносные файлы относятся к классу Ransomware.

Набор данных состоит из двух классов:

1) Класс «R» — 300000 вредоносных файлов из класса Ransomware;
2) Класс «B» — 300000 чистых легитимных приложений.

Все объекты выборки были получены из следующих источников: VirusTotal[10], VirusShare[11], Malware.lu[12], MalwareBazaar[13] и GitHib- ytisf/theZoo[14].

9 Формат PE [Электронный ресурс]. – Режим доступа: URL: https://learn.microsoft.com/ru-ru/windows/win32/debug/pe-format (дата обращения: 02.01.2024).

10 VirusTotal – Free Online Virus, Malware and URL Scanner [Электронный ресурс]. – Режим доступа: URL: https://www.virustotal.com/ (дата обращения: 02.01.2024).

11 VirusShare.com [Электронный ресурс]. – Режим доступа: URL: https://virusshare.com/ (дата обращения: 02.01.2024).

12 Malware.lu/ [Электронный ресурс]. – Режим доступа: URL: https://malware.lu/ (дата обращения: 02.01.2024).

13 MalwareBazaar | Malware sample exchange [Электронный ресурс]. – Режим доступа: URL: https://bazaar.abuse.ch/ (дата обращения: 02.01.2024).

14 A repository of LIVE malwares for your own joy and pleasure [Электронный ресурс]. – Режим доступа: URL: https://github.com[/]ytisf[/]theZoo (дата обращения: 02.01.2024).

Характеристики инструмента для формирования набора отчетов

В качестве среды для исследования поведения объектов была выбрана следующая конфигурация: Cuckoo Sandbox, продукт виртуализации Oracle VM VirtualBox с виртуальной машиной. Операционной системой была выбрана Windows 10 с предустановленными библиотеками, необходимыми для исполняемых файлов.

Данная конфигурация была выбрана по следующим причинам:

1. Возможность подключения к Cuckoo Sandbox дополнительных систем анализа;
2. Cuckoo Sandbox является open source проектом, из-за чего исходный код доступен для изучения и возможно его изменение под конкретную задачу;
3. ОС Windows 10 была выбрана в качестве гостевой операционной системы, так как это первая по числу интернет пользователей система семейства Windows[15].

Исходя из выбранного средства виртуализации и выбранной операционной системы Windows 10 множество Event состоит из 4000 событий.

Характеристики обучения DeBERTa-V3 и классификатора

С момента появления BERT многие исследовательские группы выпустили свои собственные реализации подхода, чаще всего также сопровождаемые предварительно обученными моделями. В данной работе использовалась модель DeBERTa-V3[16], которая состоит из 12 скрытых слоев, размер которых равен 768 нейронам. В качестве бинарного классификатора используется искусственная однослойная нейронная сеть (входной слой 768 нейронов, скрытый слой 768 нейронов и выходной слой размером 1 нейрон). DeBERTa-V3 и однослойная нейронная сеть обучались совместно.

Этап тонкой настройки выполняется на наборе данных, представленном в предыдущем пункте. Обучающие, валидационные и тестовые наборы распределяются как 50%, 20% и 30% соответственно. Обучающие, валидационные и тестовые наборы стратифицированы, что означает, что каждый набор имеет такое же соотношение вредоносного и чистого ПО, как и весь набор данных. Что касается параметров, то модель была точно настроена на 5 эпох с размером пакета (batch size) 32, с использованием в качестве функции оптимизатора «Adam» [18] и со скоростью обучения (learning rate) $3e^{-7}$. Эти параметры были тщательно подобраны, чтобы обеспечить наилучшую производительность модели. Все этапы обучения, тестирования и проверки выполнялись на двух графических процессорах NVIDIA GeForce RTX 3090 Ti с суммарным объёмом видеопамяти 48 ГБ.

Один полный эксперимент занял чуть меньше 13 часов. Кроме того, каждый полный эксперимент выполнялся десять раз с использованием разных исходных данных (т.е. разной последовательности для обучающих/валидационных/тестовых наборов), чтобы получить среднее значение производительности, максимально репрезентативное для модели.

Метрики оценки

В качестве мер оценок были выбраны следующие характеристики:

- ✓ Accuracy (точность) – показатель, оценивающий точность предсказания по всем классам;

$$Accurancy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- ✓ Precision – показатель, оценивающий отношение числа верно классифицируемых объектов, как положительных, к общему числу положительно распознанных объектов, правильно и неправильно;

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

- ✓ Recall – показатель, оценивающий общее отношение числа верно классифицируемых объектов к общему числу объектов в кластере;

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

- ✓ F1-мера – агрегированный показатель, объединяющий как precision, так и recall;

$$F_1 = 2\frac{precision * recall}{precision + recall} \quad (4)$$

Выбранные меры позволяют объективно оценивать результаты эксперимента.

15 Desktop Windows Version Market Share Worldwide | Statcounter Global Stats [Электронный ресурс]. – Режим доступа: URL: https://gs.statcounter.com/os-version-market-share/windows/desktop/worldwide (дата обращения: 02.01.2024).

16 The implementation of DeBERTa [Электронный ресурс]. – Режим доступа: URL: https://github.com/microsoft/DeBERTa (дата обращения: 02.01.2024).

Таблица 1

Результаты исследования в сравнении с аналогичными

Исследование	Объём выборки, шт.	Accuracy	Precision	Recall	F1 мера
Наше исследование	600 000	0.97	0.99	0.96	0.98
MalBERT	22 000	0.90	0.79	0.89	0.80
MalBERTv2	22 000	0.93	0.92	0.97	0.89

Результаты

Неудивительно, что BERT очень хорошо справляется с задачами, связанными с текстом, такими как классификация отчётов виртуализации, которые, несмотря на то что являются JSON объектами, содержат в основном текстовые данные. При этом поведенческие отчёты содержат полную информацию о поведении объектов.

Результаты исследования представлены в таблице 1. Разработанный метод имеет лучшие результаты на большем объёме данных.

Заключение

В рамках данной работы разработан метод, позволяющий с эффективностью 97% (F1 мера) распознавать программы-вымогатели на основе анализа поведенческих отчётов.

Как показано в этом исследовании, модели, обученные на поведенческих отчётах, показывают хорошие результаты. Полученные результаты интересны с точки зрения анализа вредоносного ПО на реальных системах, «в живой природе» (англ. «WildList Malware»).

Исследование проведено при финансовой поддержке Минобрнауки России («Грант ИБ МТУСИ») № 40469-25/23-К.

Литература

1. *Ijaz M., Durad M. H., Ismail M. Static and dynamic malware analysis using machine learning //2019 16th International bhurban conference on applied sciences and technology (IBCAST). – IEEE, 2019. – С. 687–691. http://dx.doi.org/10.1109/IBCAST.2019.8667136*
2. *Aboaoja F. A. et al. Malware detection issues, challenges, and future directions: A survey //Applied Sciences. – 2022. – Т. 12. – №. 17. – С. 8482.*
3. *Asghar H. J. et al. Use of cryptography in malware obfuscation //Journal of Computer Virology and Hacking Techniques. – 2024. – Т. 20. – №. 1. – С. 135–152.*
4. *Zhang X. et al. Android application forensics: A survey of obfuscation, obfuscation detection and deobfuscation techniques and their impact on investigations //Forensic Science International: Digital Investigation. – 2021. – Т. 39. – С. 301285.*
5. *Cheng B. et al. {Obfuscation-Resilient} Executable Payload Extraction From Packed Malware //30th USENIX Security Symposium (USENIX Security 21). – 2021. – С. 3451–3468*
6. *Brezinski K. et al. Metamorphic malware and obfuscation: a survey of techniques, variants, and generation kits //Security and Communication Networks. – 2021. – Т. 2023.*
7. *Alsmadi T., Alqudah N. A survey on malware detection techniques //2021 international conference on information technology (ICIT). – IEEE, 2021. – С. 371–376.*
8. *Aslan Ö. A., Samet R. A comprehensive review on malware detection approaches //IEEE access. – 2020. – Т. 8. – С. 6249–6271.*
9. *Maniriho P., Mahmood A. N., Chowdhury M. J. M. API-MalDetect: Automated malware detection framework for windows based on API calls and deep learning techniques //Journal of Network and Computer Applications. – 2023. – Т. 218. – С. 103704.*
10. *Sun J. et al. Categorizing malware via A Word2Vec-based temporal convolutional network scheme //Journal of Cloud Computing. – 2020. – Т. 9. – С. 1–14.*
11. *Chandak A., Lee W., Stamp M. A comparison of word2vec, hmm2vec, and pca2vec for malware classification //Malware analysis using artificial intelligence and deep learning. – 2021. – С. 287–320.*
12. *Yesir S., Soğukpinar İ. Malware detection and classification using fasttext and bert //2021 9th International Symposium on Digital Forensics and Security (ISDFS). – IEEE, 2021. – С. 1–6.*
13. *Pandya V. Contextualized Vector Embeddings for Malware Detection. – 2022. https://doi.org/10.31979/etd.rjra-9c8m*
14. *Rahali A., Akhloufi M. A. Malbert: Malware detection using bidirectional encoder representations from transformers //2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). – IEEE – 2021. – С. 3226–3231.*
15. *Rahali A., Akhloufi M. A. MalBERTv2: Code Aware BERT-Based Model for Malware Identification //Big Data and Cognitive Computing. – 2023. – Т. 7. – №. 2. – С. 60.*
16. *He P., Gao J., Chen W. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing // arXiv preprint arXiv:2111.09543. – 2021.*
17. *Devlin J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding //arXiv preprint arXiv:1810.04805. – 2018.*
18. *Kingma D. P., Ba J. Adam: A method for stochastic optimization //arXiv preprint arXiv:1412.6980. – 2014.*

ПРОТИВОДЕЙСТВИЕ УЯЗВИМОСТЯМ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ. Часть 2. АНАЛИТИЧЕСКАЯ МОДЕЛЬ И КОНЦЕПТУАЛЬНЫЕ РЕШЕНИЯ

Леонов Н. В.[1]

DOI: 10.21681/2311-3456-2024-3-90-95

Цель исследования: *концептуальное противодействие уязвимостям в программном обеспечении.*

Методы исследования: *системный анализ, моделирование, синтез решений.*

Полученные результаты: *во второй части статьи предложена аналитическая модель, формализуемая сущности и взаимосвязи онтологической модели области программного обеспечения с уязвимостями. Оценено влияние сущностей предметной области на реализуемость направлений в ней (программный инжиниринг, внедрение уязвимостей и их нейтрализация), что позволило синтезировать концептуальные пути противодействия уязвимостям.*

Научная новизна *работы определяется полной формализацией объектов и субъектов предметной области, а также их взаимосвязей.*

Ключевые слова: *информационная безопасность, уязвимость, противодействие, аналитическая модель, концептуальные пути решения.*

COUNTERING SOFTWARE VULNERABILITIES. Part 2. ANALYTICAL MODEL AND CONCEPTUAL SOLUTIONS

Leonov N. V.[2]

The goal of the investigation: *conceptual counteraction to software vulnerabilities.*

Research methods: *system analysis, modeling, synthesis of solutions.*

Results: *in the second part of the article, an analytical model is proposed that formalizes the essence and relationship of the ontological model of the software domain with vulnerabilities. The influence of the entities of the subject area on the feasibility of directions in it (software engineering, introduction of vulnerabilities and their neutralization) was assessed, which made it possible to synthesize conceptual ways to counter vulnerabilities.*

The scientific novelty *of the work is determined by the complete formalization of objects and subjects of the subject area, as well as their relationships.*

Keywords: *information security, vulnerability, counteraction, analytical model, conceptual solutions.*

В первой части статьи [1] введена онтологическая модель предметной области, охватившая три ее основных направления – программный инжиниринг, внедрение уязвимостей и их нейтрализация.

1 Леонов Николай Викторович, кандидат технических наук, доцент, начальник лаборатории Государственного научно-исследовательского института прикладных проблем. Москва, Россия. ORCHID: http://orcid.org/0009-0005-1295-5343. E-mail: leonov-nv@yandex.ru

2 Nikolay V. Leonov, Ph.D., Docent, Head of the State Research Institute of Applied Problems Laboratory. Moscow, Russia. ORCHID: http://orcid.org/0009-0005-1295-5343. E-mail: leonov-nv@yandex.ru

Аналитическая модель

Запишем Модель в аналитическом виде для каждого из направлений (с учетом того, что некоторые сущности относятся к нескольким направлениям) предметной области следующим образом.

Направление «Программный инжиниринг»

Получение Программы (*Program*) для решения Задачи (*Task*) Разработчиком (*Developer*) имеет следующий формальный вид:

$$Program = \widehat{Developer}^{CreateProgram}(Task),$$

где «$\widehat{\ }$» – диакритический символ «верхняя скобка» для обозначения продуцированных алгоритмов; $\widehat{Developer}^{CreateProgram}(...)$ – алгоритм создания Программы (перев. на англ. *Create Program*), продуцируемый Разработчиком.

В рамках Модели создание Программы с помощью Средства сборки (*BuildTool*) [2], применяемого Разработчиком, имеет следующий формальный вид:

$$Program^{ExecutableCode} = BuildTool(Program^{SourceCode}, \widehat{DeveloperAlgorithm})^{BuildTool},$$

где $Program^{ExecutableCode}$ и $Program^{SourceCode}$ – Программа в представлении Исполняемого и Исходного кода [3]; $BuildTool(...)$ – алгоритм работы Средства сборки; $\widehat{DeveloperAlgorithm}^{BuildTool}$ – алгоритм по управлению Средством, продуцируемый (*Production*) Разработчиком, т.е.:

$$\widehat{DeveloperAlgorithm}^{BuildTool} := Developer^{Production}(GoalBuilding^{Program}),$$

где «:=» – обозначение назначения алгоритма; $Developer^{Production}(...)$ – продуцирование нового алгоритма Разработчиком; $Goal^{BuildingProgram}$ – цель (перев. на англ. *Goal*) по сборке Программы (перев. на англ. *Build Program*), достигаемая Разработчиком.

Направление «Нейтрализация уязвимостей»

Получение Отчета безопасности (*SecurityReport*) [4] путем анализа Программы Средством сканирования (*ScannerTool*) [5], применяемым Экспертом (*Expert*) [6], имеет следующий формальный вид:

$$SecurityReport = ScannerTool(Program, \widehat{ExpertAlgorithm}^{ScannerTool}),$$

где $ScannerTool(...)$ – алгоритм работы Средства сканирования; $\widehat{ExpertAlgorithm}^{ScannerTool}$ – алгоритм по управлению Средством, продуцируемый Экспертом на основании Мета-информации ($MetaInformation^{Expert}$) [7], т.е.:

$$\widehat{ExpertAlgorithm}^{ScannerTool} := Expert^{Production}(Goal^{ScanningProgram}, MetaInformation^{Expert}),$$

где $Expert^{Production}(...)$ – продуцирование нового алгоритма Экспертом; $Goal^{ScanningProgram}$ – цель по сканированию Программы (перев. на англ. *Scan Program*), достигаемая Экспертом.

Получение Патча (*Patch*) [8] по Отчету безопасности путем действий Эксперта имеет следующий формальный вид:

$$Patch = \widehat{Expert}^{CreatePatch}(Task, MetaInformation^{Expert}),$$

где $\widehat{Expert}^{CreatePatch}(...)$ – алгоритм создания Патча (перев. на англ. *Create Patch*), продуцируемый Экспертом на основании Мета-информации.

Внедрение Патча в Программу с помощью Средства внедрения (*EmbeddingTool*), применяемого Экспертом, имеет следующий формальный вид:

$$Program = EmbeddingTool(Patch, \widehat{ExpertAlgorithm}^{EmbeddingTool}),$$

где $EmbeddingTool(...)$ – алгоритм работы Средства внедрения; $\widehat{ExpertAlgorithm}^{EmbeddingTool}$ – алгоритм по управлению Средством, продуцируемый Экспертом на основании Мета-информации, т.е.:

$$\widehat{ExpertAlgorithm}^{EmbeddingTool} := Expert^{Production}(Goal^{PatchingProgram}, MetaInformation^{Expert}),$$

где $Goal^{PatchingProgram}$ – цель по исправлению Программы (перев. на англ. *Patching Program*) с позиции имеющихся в ней ошибок, достигаемая Экспертом.

Запутывание кода Программы с помощью Средства обфускации (*ObfuscationTool*) [9], применяемого Экспертом, имеет следующий формальный вид:

$$Program' = ObfuscationTool(Program, \widehat{ExpertAlgorithm}^{ObfuscationTool}),$$

где «′» – диакритический символ указания новой версии объекта (в данном случае Программы); $Program'$ – Программа после обфускации; $ObfuscationTool(...)$ – алгоритм работы Средства обфускации [10]; $\widehat{ExpertAlgorithm}^{ObfuscationTool}$ – алгоритм по управлению Средством, продуцируемый Экспертом на основании Мета-информации, т.е.:

$$\widehat{ExpertAlgorithm}^{ObfuscationTool} := Expert^{Production}(Goal^{ObfuscatingProgram}, MetaInformation^{Expert}),$$

где $Goal^{ObfuscatingProgram}$ – цель по обфускации Программы (перев. на англ. *Obfuscating Program*), достигаемая Экспертом.

Направление «Внедрение уязвимостей»

Получение Уязвимости (*Vulnerability*) для осуществления Вектора атаки (*AttachVector*) Нарушителем (*Intruder*) [11] имеет следующий формальный вид:

$$Vulnerability = \widehat{Intruder}^{CreateVulnerability}(AttachVector, MetaInformation^{Intruder}),$$

где $\widehat{Intruder}^{CreateVulnerability}(...)$ – алгоритм создания Уязвимости (перев. на англ. *Create Vulnerability*), продуцируемый Нарушителем на основании Мета-информации ($MetaInformation^{Intruder}$).

Запутывание кода Уязвимости с помощью Средства обфускации, применяемого Нарушителем [12], имеет следующий формальный вид:

$$Vulnerability' = ObfuscationTool(Vulnerability, \widehat{IntruderAlgorithm}^{ObfuscationTool}),$$

где $Vulnerability'$ – Уязвимость после обфускации; $\widehat{IntruderAlgorithm}^{ObfuscationTool}$ – алгоритм по управлению Средством, продуцируемый Нарушителем на основании Мета-информации, т.е.:

$$\widehat{IntruderAlgorithm}^{ObfuscationTool} := Intruder^{Production}(Goal^{ObfuscatingVulnerability}, MetaInformationI^{Intruder}),$$

где $Goal^{ObfuscatingVulnerability}$ – цель по обфускации Уязвимости (перев. на англ. *Obfuscating Vulnerability*), достигаемая Нарушителем.

Внедрение Уязвимости [13, 14] в Программу с помощью Средства внедрения, применяемого Нарушителем, имеет следующий формальный вид:

$$Program = EmbeddingTool(Vulnerability, \widehat{IntruderAlgorithm}^{EmbeddingTool}),$$

где $\widehat{IntruderAlgorithm}^{EmbeddingTool}$ – алгоритм управления Средством внедрения, продуцируемый Нарушителем на основании Метаинформации, т.е.:

$$\widehat{IntruderAlgorithm}^{EmbeddingTool} := Intruder^{Production}(Goal^{InfectingProgram}, MetaInformation^{Intruder}),$$

где $Goal^{InfectingProgram}$ – цель по заражению Программы (перев. на англ. *Infecting Program*), достигаемая Экспертом.

В данном случае применение алгоритма Средства внедрения с формальной точки зрения одинаково как Экспертом для Патча, так и Нарушителем для Уязвимости, поскольку у алгоритма $EmbeddingTool(...,...)$ 1-м параметром идет некоторый код (*Patch* или *Vulnerability*), а вторым – алгоритм управления средством ($ExpertAlgorithm^{EmbeddingTool}$ или $IntruderAlgorithm^{EmbeddingTool}$).

Получение Метаинформации из Программы с помощью Средства реинжиниринга ($ReengineeringTool$) [15, 16], применяемого Экспертом и Нарушителем, имеет следующий формальный вид:

$$\begin{cases} MetaInformation^{Expert} = \text{ReengineeringTool}(\text{Program}, \widehat{ExpertAlgorithm}^{ReengineeringTool}) \\ MetaInformation^{Intruder} = \text{ReengineeringTool}(\text{Program}, \widehat{IntruderAlgorithm}^{ReengineeringTool}) \end{cases},$$

где $ReengineeringTool(...)$ – алгоритм работы Средства реинжиниринга (перев. на англ. $ReengineeringTool$) [17]; $Expert^{ReengineeringTool}$ и $Intruder^{ReengineeringTool}$ – алгоритмы Эксперта и Нарушителя по управлению средством для получения собственной метаинформации $MetaInformation^{Expert}$ и $MetaInformation^{Intruder}$, соответственно, т.е.:

$$\begin{cases} \widehat{ExpertAlgorithm}^{ReengineeringTool} := Expert^{Production}(Goal^{ReverseEngineering}, MetaInformation^{Expert}) \\ \widehat{IntruderAlgorithm}^{ReengineeringTool} = Intruder^{Production}(Goal^{ReverseEngineering}, MetaInformation^{Intruder}) \end{cases},$$

где $Goal^{ReverseEngineering}$ – цель по реверс-инжинирингу Программы (перев. на англ. *Reverse Engineering*), достигаемая Экспертом и Нарушителем в ходе собственной, диаметрально противоположной деятельности.

Корректность и охват записи Модели в аналитическом виде позволяет сделать следующие утверждения.

Во-первых, поскольку результаты каждой аналитической записи используется в какой-либо другой, то, следовательно, все сущности Модели являются необходимыми; иначе, если бы в Модели присутствовала некоторая неиспользуемая сущность, то одна из записей возвращала бы результат, не используемый в других. Например, если бы Эксперт в результате анализа Программы (предположим, Средством сканирования) создавал бы ее электронную подпись – которая, очевидно, бессмысленна для нейтрализации уязвимостей – то присутствовала бы аналитическая запись, результатом которой являлась эта подпись, не используемая ни в каких других записях.

Во-вторых, поскольку для каждой аналитической записи введены или получены используемые в ней параметры, то, следовательно, все сущности Модели являются достаточными; иначе, если бы в Модели отсутствовала некоторая сущность, то и одна из записей была бы невычислима из-за отсутствия необходимого параметра. Например, если для Эксперта отсутствовал бы А-объект – Мета-информация, то создание Экспертом Патча для конкретной Программы (с помощью $\widehat{Expert}^{CreatePatch}$ было бы невозможно из-за отсутствия информации о принципах и деталях ее работы (т.е. $MetaInformation^{Expert}$).

Таким образом, можно утверждать, что все сущности Модели являются необходимыми и достаточными.

Направление «Нейтрализация уязвимостей»

Исходя из того, что три направления предметной области (Программный инжиниринг, Нейтрализация и Внедрение уязвимостей) построены на общем терминологическом базисе, при этом в единой нотации (используя жестко заданные и ограниченные субъекты, объекты и их связи), можно сделать следующее предположение:

«Влияние, оказываемое на отдельные сущности, будет оказывать влияние и на реализуемость направлений».

Следовательно, снижение эффективности действий Нарушителя может быть достигнуто через влияние на сущности, с которыми он «работает»; сложность такого противодействия уязвимостям заключается и в потенциальном обратном влиянии на другие, легальные направления. Например, достаточно смелое «удаление» из онтологической модели основополагающей сущности Программы, как совокупности Исходного и Исполняемого кода со средством его сборки (т.е. некий «фантастический» сценарий полного отказа от информационных технологий), хотя и не позволит Нарушителю закладывать в нее Уязвимости и проводить Вектор атаки (при автоматическом избавлении Эксперта от необходимости анализа Программы и создания для нее Патча), тем не менее и Разработчик не сможет решать Задачу программным способом. Примером менее радикального способа является запрет использования или строгий контроль любых средств модификации Программ (таких, как Средство внедрения), что хотя и не позволит исправлять случайные ошибки Разработчиков с помощью Патчей, но и существенно затруднит Нарушителям встраивание вредоносного кода. Связь каждого объекта Модели с реализуемостью каждого из направлений приведены в табл. 1, где введены следующие обозначения степени влияния: «+» – необходимость сущности для направления (зеленый фон); «–» – отсутствие необходимости (синий фон); «+/–» – повышение эффективности его реализуемости (белый фон). Подобные связи для субъектов будут опущены, поскольку их наличие влияет исключительно на каждое из направлений. Влияние же взаимодействий между сущностями сложно анализируемо и контролируемо (например, при участии трех сущностей).

Дадим ряд пояснений касательно выставленных степеней влияния каждой сущности на направления и качественных способов противодействия путем управления ее (см. табл. 1) – т.е. повышения эффективности нейтрализации Уязвимостей и/или снижения эффективности внедрения Уязвимостей с сохранением направления Программного инжиниринга.

Во-первых, наличие «+» только для одного из направлений тождественно отсутствию влияния сущности на другие направления. Таким образом, исключение Вектора атаки и Уязвимости из предметной области (т.е. их полная нейтрализация) позволяет противодействовать Внедрению уязвимостей, для чего, впрочем, уже существует достаточное количество программных, технических, организационных и иных решений. Соответственно, подобные («однонаправленные») сущности (Задача, Отчет безопасности, Патч, Средство сканирования) для своих направлений не должны быть исключены или ограничены.

Во-вторых, наличие «+» для двух и более направлений тождественно необходимости наличия сущностей во всех из них; это верно для Программы. Таким образом, нейтрализация Исполняемого кода привела бы и к противодействию направления «Программная инженерия», что, естественно, недопустимо.

Таблица 1

Влияние сущностей предметной области на реализуемость в ней направлений

Сущность	Направление			Способ противодействия
	Программный инжиниринг	Нейтрализация уязвимостей	Внедрение уязвимостей	
Задача	+	–	–	Отсутствует
Исходный код	+	+/–	+/–	Ограничение
Исполняемый код	+	+	+	Ограничение
Вектор атаки	–	–	+	Исключение
Уязвимость	–	–	+	Исключение
Отчет безопасности	–	+	–	Отсутствует
Патч	–	+	–	Отсутствует
Метаинформация	–	+	+	Ограничение
Средство сборки	+	+/–	+/–	Ограничение
Средство сканирования	–	+	–	Отсутствует
Средство внедрения	–	+	+	Ограничение
Средство обфускации	–	+/–	+/–	Исключение Ограничение
Средство реинжиниринга	–	+/–	+	Исключение Ограничение

Нейтрализация же Метаинформации и Средства внедрения привела бы к двойному эффекту. Так, как Нарушитель не смог бы анализировать код, разрабатывать под него Уязвимости и встраивать их в Программу, так и Эксперт остался бы без Отчетов безопасности и возможности применения Патчей. Однако, в этом случае возможным способом противодействия может стать ограничение доступа к сущности со стороны Нарушителя, оставляя при этом полный доступ для Разработчика и Эксперта. Так, если ограничение доступа Нарушителя к Программе представляется сложно осуществимым, то препятствование распространению Метаинформации о различных популярных или используемых в критических областях Программах является более чем разумным и возможным; Средства внедрения и вовсе могут быть отнесены к разрешенным для использования только в специальных организациях, занимающихся вопросами информационной безопасности. Исключение Исполняемого кода из предметной области, как очевидно, привела бы к полному прекращению функционирования любого ПО, а меры ограничения доступа к нему со стороны Нарушителя являются практически неосуществимыми на практике.

В-третьих, наличие «+/–» для нескольких направлений указывает на желательность, но не на необходимость данной сущности; так, применение Средства обфускации носит как положительный эффект – снижает эффективность внедрения Уязвимости в Программу, так и отрицательный – затрудняет поиск этой Уязвимости. Таким образом, исключение Средства обфускации из предметной области неоднозначно (а скорее – несущественно) повлияет на безопасность Программы. Впрочем, ограничение доступа Нарушителям к средствам затруднения анализа кода Программы при наличии такового для Эксперта даст преимущество последнему.

И, в-четвертых, наличие «+» для одного направления и «+/–» для другого означает повышение эффективности второго при необходимости для первого. Соответственно, исключение данной сущности позволит оказать качественное противодействие первому направлению, при этом лишь снизив эффективности реализуемости второго. Так, исключение из предметной области Средства реинжиниринга существенно затруднит Нарушителю внедрение Уязвимости в Программу (поскольку, получение им Метаинформации будет невозможно, а «слепое» изменение Исполняемого кода скорее всего приведет к нарушению функционирования); при этом без такого Средства Эксперт также сможет получать Отчеты безопасности и внедрять Патчи, хотя и с меньшей эффективностью (поскольку, Разработчики также заинтересованы в устранении Уязвимости и будут более активно взаимодействовать с Экспертами). Аналогично предыдущему пояснению, ограничение доступа к Средствам реинжиниринга существенно затруднит анализ Нарушителем Программы при неизменности подобного анализа для Эксперта, что является наиболее предпочтительным вариантом. С другой стороны, исключение Исходного кода и Средства сборки хотя и снизит эффективность внедрения и нейтрализации Уязвимости, однако решение Задач с применением Программ окажется практически невозможным.

Исходя из сделанных пояснений, можно предложить следующие, достаточно строго полученные, предпосылки, которые должны лечь в основу соответствующих концептуальных путей противодействия Уязвимостям в Программах.

1) Вектор атаки и Уязвимость должны быть исключены из предметной области;
2) Доступ к Исходному коду, Метаинформации, а также Средствам сборки и внедрения должны быть ограничены для Нарушителя;
3) Средства обфускации и реинжиниринга могут быть исключены из предметной области, однако более целесообразным вариантом должно быть ограничение доступа к ним Нарушителя;
4) Остальные сущности (Исполняемый код, Задача, Отчет безопасности, Патч и Средство сканирования) должны присутствовать в предметной области.

Естественно, для отражения реального состояния дел в предметной области требуется более «тонкая» (по возможности – количественная) оценка влияния сущностей на направления, а также способов воздействия на них. Так, Средство реинжиниринга неявно присутствует в любой работе Эксперта (например, для связи областей Исходного или Исполняемого кода Программы с Уязвимостями в Отчете безопасности), что также требуется учитывать.

Заключение

В работе рассмотрена основополагающая проблема противодействия уязвимостям ПО с позиции трех действующих «игроков» сферы информационной безопасности – Разработчика, Нарушителя и Эксперта. Для этого введена онтологическая модель предметной области – разработка Программы, внедрение в нее уязвимостей и их нейтрализации, – построенная по единому (шаблонированному) принципу связей ее сущностей и являющаяся основным научным результатом.

Частный научный результат состоит в таблице влияния сущностей предметной области на реализуемость ее основных направлений, а также в предложенных качественных способах противодействия.

Новизна основного научного результата заключается в использовании строгих правил (или шаблонов) связей между ее сущностями предметной

области, а также в представлении модели в полностью аналитическом виде.

Теоретическая значимость состоит в установлении строгой взаимосвязи между Разработчиком программы, внедряющим в нее уязвимости Нарушителем, и противодействующим уязвимостям Экспертом посредством общих (т.е. разделяемых) сущностей предметной области.

Практическая значимость состоит в возможности выработки концептуальных способов влияния на сущности предметной области (путем их исключения или ограничения доступа) в интересах решения глобальной задачи обеспечения безопасности ПО.

В качестве продолжения работы планируется детализация представленной Модели, добавление новых противодействующих друг другу участников, более тонкий учет связей между сущностями Модели, а также разработка конкретных путей противодействия Уязвимостям в Программах.

Литература

1. *Леонов Н. В. Противодействие уязвимостям программного обеспечения. Часть 1. Онтологическая модель // Вопросы кибербезопасности. № 2(60). 2024. DOI: 10.21681/2311-3456-2024-2-87-92*
2. *Миронов С. В., Батраева И. А., Дунаев П. Д. Библиотека для разработки компиляторов // Труды Института системного программирования РАН. 2022. Т. 34. № 5. С. 77–88. DOI: 10.15514/ISPRAS-2022-34(5)-5.*
3. *Афонин М. В. Компиляция. Сборка и связывание проектов // Инновационный потенциал развития общества: взгляд молодых ученых: сборник научных статей 3-й Всероссийской научной конференции перспективных разработок (Курск, 01 декабря 2022 года). Том 3. 2022. С. 115–118.*
4. *Якимук А. Ю., Устинов С. А., Лазарев Т. П., Коваленко А. С. Методы формализации описания сценариев кибератак // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. 2022. № 1–2. С. 73–76.*
5. *Суздалов Д. В., Некрасов А. Н. Разработка сканера уязвимостей // Наука молодых: сборник материалов Межрегиональной молодежной научной конференции, посвященной памяти Ф. А. Бабушкина, (Сыктывкар, 25–26 мая 2023 года). 2023. С. 139–143.*
6. *Вареница В. В., Марков А. С., Савченко В. В., Цирлов В. Л. Практические аспекты выявления уязвимостей при проведении сертификационных испытаний программных средств защиты информации // Вопросы кибербезопасности. 2021. № 5 (45). С. 36–44. DOI: 10.21681/2311-3456-2021-5-36-44.*
7. *Израилов К. Е. Методология реверс-инжиниринга машинного кода. Часть 2. Статическое исследование. Труды учебных заведений связи // 2023. Т. 9. № 6. С. 68–82. DOI: 10.31854/1813-324X-2023-9-6-68-82.*
8. *Коржев А. А. Обеспечение безопасности программного обеспечения // Стратегическое развитие инновационного потенциала отраслей, комплексов и организаций: сборник статей XI Международной научно-практической конференции (Пенза, 10–11 октября 2023 года). 2023. – С. 237–241.*
9. *Градский Д. Ю. Методы обфускации кода // Оригинальные исследования. 2020. Т. 10. № 5. С. 177–180.*
10. *Иванов М. А., Коннова И. Г., Саликов Е. А., Степанова М. А. Обфускация логических схем генераторов псевдослучайных чисел на регистрах сдвига с линейными и нелинейными обратными связями // Безопасность информационных технологий. 2021. Т. 28. № 1. С. 74–83. DOI: 10.26583/bit.2021.1.06.*
11. *Лукацкий А. В. Обзор мировых трендов по промышленной кибербезопасности // Релейщик. 2020. № 1 (36). С. 60–62.*
12. *Ерохин В. В., Притчина Л. С. Анализ и совершенствование методов обнаружения шелл-кодов (shellcode) в компьютерных системах // Прикладная информатика. 2021. Т. 16. № 2 (92). С. 103–122.*
13. *Руднев Н. О., Герасимова В. Ф., Шагапов И. А. Метод закрепления доступа в системе посредством инъекции кода в операционной системе Windows // Естественные и технические науки. 2022. № 12 (175). С. 398–403.*
14. *Нефедов В. В. Методы внедрения кода в исполняемые файлы PE-формата // Молодежная научная школа кафедры «Защищенные системы связи». 2021. Т. 1. № 2 (4). С. 61–68.*
15. *Маркин Д. О., Макеев С. М. Система защиты терминальных программ от анализа на основе виртуализации исполняемого кода // Вопросы кибербезопасности. 2020. № 1 (35). С. 29–41. DOI: 10.21681/2311-3456-2020-01-29-41.*
16. *Буйневич М. В., Ганов Г. А., Израилов К. Е. Интеллектуальный метод визуализации взаимодействий программ в интересах аудита информационной безопасности операционной системы // Информатизация и связь. 2020. № 4. С. 67–74.*
17. *Фомин А. И. Оценка сложности исследования дизассемблированного кода исполняемых программ // Естественные и технические науки. 2021. № 7 (158). С. 210–211.*

АСИМПТОТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ ОТКРЫТОГО СЕТЕВОГО КЛЮЧЕВОГО СОГЛАСОВАНИЯ

Синюк А. Д.[1], Потапов И. А.[2], Остроумов О. А.[3]

DOI: 10.21681/2311-3456-2024-3-96-104

Цель исследования *– поиск путей уменьшения времени восстановления сетевой криптосвязности.*

Метод исследования *– введение в теорию информации коэффициента асимптотического выигрыша по времени согласования сетевого ключа в условиях неограниченного увеличения длины передаваемой последовательности и заданных требований к открыто формируемому ключу сети связи.*

Результаты исследований *– исследуются две модели открытого формирования ключа. В первой модели первоначально поочередно формируются ключи в каждом канале сети связи, а затем один из корреспондентов выбирает один из ключей в качестве сетевого и передает его по закрытым каналам другим корреспондентам. Во второй - ключ формируется одновременно по составляющим каналам сети. Поэтому вводится коэффициент асимптотического выигрыша по времени формирования ключа трех сетевых корреспондентов определяющий показатель асимптотической эффективности открытого сетевого ключевого согласования. Выполнена оценка показателя эффективности, позволившая найти преимущественные теоретико-информационные условия использования каждой из моделей.*

Практическая ценность *– результаты могут быть полезны исследователям для анализа различных подсистем информационной безопасности телекоммуникационных систем для оценки потенциальных возможностей по уменьшению времени восстановления криптосвязности.*

Ключевые слова: *теория информации; сеть связи; нарушитель; сетевой ключ; ключевая пропускная способность; коэффициент асимптотического выигрыша по времени формирования сетевого ключа.*

ASYMPTOTIC EFFICIENCY OF OPEN NETWORK KEY CONNECTION

Sinyuk A. D.[4], Potapov I. A.[5], Ostroumov O. A.[6]

Abstract: Maintaining. *The key management subsystem main function of a telecommunication system in the key compromises context by an intruder is to ensure cryptographic connectivity timely restoration of geographically dispersed correspondents via secure channels, which is updated for network correspondents, due to the fact that the resistance of the network key to compromise is minimal.*

The study purpose *is to find ways to reduce the recovery time of network crypto-connectivity.*

The research method *is the introduction into information theory of the coefficient of asymptotic gain in time of network key agreement under conditions of an unlimited increase in the length of the transmitted sequence and specified requirements for an openly generated key of a communication network.*

Research results *– two models of open key generation are investigated. In the first model, keys are initially generated in turn in each channel of the communication network, and then one of the correspondents*

1 Синюк Александр Демьянович, доктор технических наук, доцент, профессор кафедры Общепрофессиональных дисциплин Военной орденов Жукова и Ленина краснознаменной академии связи имени Маршала Советского Союза С. М. Буденного, Санкт-Петербург, Россия. E-mail: eentrop@rambler.ru, orcid.org/0000-0003-0608-4359.

2 Потапов Илья Александрович, кандидат технических наук, доцент, доцент кафедры Общепрофессиональных дисциплин Военной орденов Жукова и Ленина краснознаменной академии связи имени Маршала Советского Союза С.М. Буденного, Санкт-Петербург, Россия. E-mail: momento87@mail.ru

3 Остроумов Олег Александрович, кандидат технических наук, старший преподаватель кафедры Военных систем многоканальной, электропроводной и оптической связи Военной орденов Жукова и Ленина краснознаменной академии связи имени Маршала Советского Союза С. М. Буденного, Санкт-Петербург, Россия. E-mail: oleg-26stav@mail.ru, orcid.org/0000-0003-1674-6248.

4 Alexander D. Sinyuk, Dr.Sc., Associate Professor, Professor of the Department of General Professional Disciplines of the Military Orders of Zhukov and Lenin of the Red Banner Academy of Communications named after Marshal of the Soviet Union S.M. Budyonny, St. Petersburg, Russia. E-mail:eentrop@rambler.ru, orcid.org/0000-0003-0608-4359..

5 Ilya A. Potapov, Ph.D., Associate Professor, Associate Professor, Department of General Professional Disciplines, Military Orders of Zhukov and Lenin, Red Banner Academy of Communications named after Marshal of the Soviet Union S.M. Budyonny, St. Petersburg, Russia. E-mail:momento87@mail.ru

6 Oleg A. Ostroumov, Ph.D., senior lecturer of the Department of Military Systems of Multichannel, Electrically Conducted and Optical Communications of the Military Orders of Zhukov and Lenin of the Red Banner Academy of Communications named after Marshal of the Soviet Union S.M. Budyonny, St. Petersburg, Russia. E-mail: oleg-26stav@mail.ru, orcid.org/0000-0003-1674-6248.

selects one of the keys as a network key and transmits it through private channels to other correspondents. In the second, the key is formed simultaneously along the constituent channels of the network. Therefore, a coefficient of asymptotic gain in the time of key formation of three network correspondents is introduced, which is a determining indicator of open network key agreement asymptotic efficiency. An assessment of the efficiency indicator was carried out, which made it possible to find the preferential information-theoretic conditions for using each of the models.

***Practical value** – the results can be useful to researchers for analyzing various information security subsystems of telecommunication systems to assess the potential for reducing the recovery time of crypto-connectivity.*

***Discussion.** The results obtained deepen and expand the known information-theoretic assessments of various key coordination models effectiveness.*

***Keywords:** information theory; communication network; intruder; network key; open network key negotiation; key throughput; coefficient of asymptotic gain in the time of network key formation.*

Введение

Современные защищенные телекоммуникационные системы включают в своем составе подсистемы управления криптографическими ключами. Главной целью в условиях компрометаций ключей нарушителем любой подсистемы управления ключами выступает решение задачи своевременного восстановления по защищенным каналам криптосвязности территориально разнесенных корреспондентов (объектов связи), которое актуализируется для сетевых корреспондентов, ввиду того, что устойчивость сетевого ключа к компрометациям минимальна. Это актуализирует поиск путей достижения потенциальных возможностей по уменьшению времени восстановления криптографической связности объектов связи после компрометации сетевого ключа нарушителем.

В предыдущих исследованиях [1] показано, что в условиях одновременной передачи информации одновременно по составляющим каналам дискретного широковещательного канала связи без памяти (ДШКБП), может быть затрачено меньше времени на передачу, чем при поочередной передаче информации по каждому составляющему каналу ДШКБП. Этот факт определяет возможный выигрыш по времени формирования сетевого ключа (СК) объектов связи (ОС) на основе открытого сетевого ключевого согласования [2, 3]. Поэтому в работе предлагается терминология и метод оценки асимптотической эффективности открытого сетевого ключевого согласования, который позволит обоснованно выбирать преимущественные условия осуществления формирования СК по открытым каналам сети связи, обеспечивающие оперативное восстановление сетевой криптосвязности. Вводится коэффициент асимптотического выигрыша по времени формирования СК в условиях неограниченного увеличения длины передаваемого открытого сообщения (кодового слова) [4] и заданных требований к СК [2].

Полученные результаты расширяют область известных исследований открытого ключевого согласования и могут быть использованы для оценки потенциальных возможностей и анализа предлагаемых современных криптографических подсистем защиты информации телекоммуникационных систем [5], включающих подсистемы управления ключами.

Предварительные результаты

В источниках [2] исследована модель формирования ключа для трех сетевых объектов связи (ОС). Рассмотрено следующее общее описание ситуации передачи информации в сети связи (по широковещательному каналу связи), показанной на рис. Имеется один передатчик (кодер) у ОС $\boldsymbol{A}$ и три независимо работающих приемника (декодера) у ОС $\boldsymbol{B}$, $\boldsymbol{C}$ и нарушителя $\boldsymbol{E}$, на входы которых поступают выходные сигналы разных каналов.

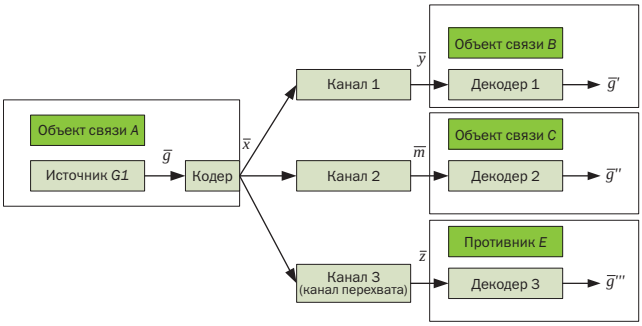


Рис. 1. Модель сетевой канальной связности ОС A, B, C и нарушителя E

На передатчик поступают сообщения $\bar{g}$ от источника $G1$ (который находится у ОС A), которые он должен передать в одно и то же время приемникам 1, 2 и 3 (ОС B, C и нарушителю E, соответственно) так, чтобы приемники 1 и 2 могли восстановить с произвольно малой вероятностью ошибки сообщения источника $G1$, а нарушитель не был в состоянии восстановить с произвольно малой вероятностью ошибки сообщения источника $G1$. Совокупность, состоящая из источника сообщений $G1$ и кодера, приемников, каналов связи образует модель канальной связности (МКС).

Пусть ОС A использует для генерации сетевого ключа (СК) дискретный источник без памяти $G1$ [6] модели канальной связности трех сетевых ОС и нарушителя с равномерным законом распределения вероятностей сообщений [7].

Все каналы связи в МКС описываются моделями дискретных симметричных каналов связи без памяти (ДСК) [8]. Совокупность, состоящая из двух каналов с общим входом (выход кодера, который находится у ОС A) и выходами (входы приемников 1, 2, которые находятся у ОС B, C, соответственно) описывается моделью дискретного широковещательного канала без памяти (ДШКБП), который был исследован [9].

Передача сигналов по ДШКБП определяется двумя каналами с общим входным алфавитом X, выходными алфавитами Y и M и матрицами переходных вероятностей $P_1 = \{p(y/x)\}$, $P_2 = \{p(m/x)\}$, $x \in X$, $y \in Y$, $m \in M$. Алфавиты X, Y и M конечны и для любых последовательностей $\bar{x} \in X^n$, $\bar{y} \in Y^n$, $\bar{m} \in M^n$, где X^n – декартова n-я степень множества X [6], а Y^n и M^n определяются аналогично. Дискретный широковещательный канал без памяти обозначим символом $\{X,Y,M;p(y/x),p(m/x)\}$, при этом каналы $\{X,Y;p(y/x)\}$ и $\{X,M;p(m/x)\}$ являются составляющими ДШКБП. Канал связи с входом на выходе кодера, который находится у ОС A, и с выходом на входе приемника 3, который находятся у нарушителя E определен в [2] как канал перехвата (КП). Передача сигналов по КП (описывается моделью дискретного симметричного канала связи без памяти) определяется входным алфавитом X, выходным алфавитом Z и матрицей переходных вероятностей $P_3 = \{p(z/x)\}$, $x \in X$, $z \in Z$. Для КП алфавиты X, Z конечны и для любых последовательностей $\bar{x} \in X^n$, $\bar{z} \in Z^n$, где X^n – декартова n-я степень множества X и Z^n – декартова n-я степень множества Z. КП обозначим символом $\{X,Z;p(z/x)\}$.

Составляющие ДШКБП и КП являются независимыми каналами [8]. Алфавиты (объемы алфавитов) источника $G1$, алфавит, описывающий вход ДШКБП и КП, выходные алфавиты составляющих ДШКБП и КП совпадают, т. е. $|G| = |X| = |Y| = |M| = |Z| = t$.

Для передачи информации используется случайный кодер, математическое описание которого представлено в [2]. Нарушитель E при использовании случайного кодера представляется случайным выбором сообщения источника $G1$ (это соответствует случайному выбору кодового подмножества) и равномерным распределением входной последовательности на входе КП при условии известного выбора сообщения источника $G1$.

Общая постановка задачи формирования общего ключа (СК) трех сетевых объектов связи по открытым каналам связи с ошибками сводится к необходимости выработать общий ключ (СК) для сети из трех сетевых ОС, передавая данные по ДШКБП и КП. Требуется обеспечить формирование общего СК с высокой надежностью для ОС и обеспечить наперед заданный низкий уровень информации об этом СК со стороны нарушителя. Предполагается, что нарушитель использует пассивную стратегию [2, 10]. Особенности активного нарушителя показаны в [11, 12]. После передачи информации ОС обладают некоторой информацией в виде последовательностей на входе ДШКБП и двух его выходах в виде кодового слова $\bar{x}$, $\bar{x} \in V$ для ОС A и принятых последовательностей $\bar{y}$, $\bar{y} \in Y^n$ для ОС B и $\bar{m}$, $\bar{m} \in M^n$ для ОС C. Эти последовательности могут быть коррелированы между собой [13], а также с начальными данными нарушителя в виде последовательности $\bar{z}$, $\bar{z} \in Z^n$. Предполагается, что нарушитель E знает полное описание всех действий выполняемых ОС, выбранного кода (n,ε_1) и источника $G1$, как и в модели [2, 14]. Первоначально распределенные (переданные) последовательности не могут быть использованы для формирования СК, т.к. в ДШКБП (в составляющих ДШКБП) могут возникать ошибки. Тогда они требуют коррекции с использованием метода декодирования полученных на выходах ДШКБП последовательностей. После чего декодированные последовательности и первоначально выбранная последовательность источника $G1$ могут быть выбраны в качестве СК для ОС B, C и A. Эти условия определяют построение для достаточно большого n протокола формирования общего СК, который приведен в [2].

Основные показатели качества, сформированного (переданного) СК после использования ОС протокола формирования общего СК для достаточно большого n приведены в [2]. Качественно они сводятся к надежной передаче большого количества бит «хорошего» СК при малой утечке информации к нарушителю E. «Информационная» скорость формирования СК по отношению к длине кодового слова (КС) n характеризует оперативность установления криптосвязности между ОС. СК удовлетворяет ряду требований, показанных в [2].

Определена ключевая пропускная способность для трех сетевых ОС C_3 [2] как максимально достижимая величина скорости формирования ключа для трех сетевых ОС H_3. Разработанная модель канальной связности сетевых ОС и нарушителя, приведенная на рис. 1, выполнение во времени ОС шагов протокола формирования СК для формирования общего СК трех сетевых ОС представляется как процесс формирования СК для трех объектов связи во времени.

Основным параметром, характеризующим быстротечность процесса формирования СК для сетевых объектов связи (ПФСК), является C_3 – ключевая

пропускная способность для трех сетевых ОС, которая достигается при $n \to \infty$ и равномерном распределении на входе ДШКБП и КП в силу свойств информационных мер дискретных симметричных каналов, приведенных в [5, 8], и совсем не зависит от модели, описывающей источник $G1$. Показатель C_3 зависит от потенциальных возможностей ДШКБП и КП. Пропускная способность ДШКБП $C_{ДШКБП}$ определяет (измеряет) потенциальное (предельное) количество информации источника, которое может нести один символ кодового слова [15]. Это же можно утверждать относительно пропускной способности КП C_w. А так как теория информации [16] только измеряет количество информации, то только превышение $C_{ДШКБП}$ над C_w гарантирует передачу по ДШКБП по сравнению с КП той положительной разницы в количестве информации, которая в последующем может быть использована для формирования СК.

Если один из составляющих каналов ДШКБП не подвержен влиянию ошибок (помех), то C_3 определяется как секретная пропускная способность с каналом перехвата C_S для двух ОС, в МКС, предложенной Чисаром и Кернером [5], в которой основной канал (канал между ОС) и канал перехвата имеют общий вход и являются независимыми каналами [9, 17].

Постановка задачи

В [1] показано, что один из возможных методов передачи по ДШКБП состоит в разделении времени, когда в течение некоторого отрезка времени осуществляется передача одному приемнику, а в течение другого отрезка времени – второму. Другая возможность состоит в том, чтобы вести передачу обоим приемникам в одно и то же время. Первый случай широко исследован в [18]. В статье исследуется последний случай. Процесс формирования СК основан на передаче сообщений. Тогда в рамках МКС возможны 2 варианта формирования СК. В первом случае СК может формироваться при поочередной передаче информации по каждому составляющему каналу ДШКБП, который соответствует случаю разделения времени и формированию СК в модели Чисара и Кернера [5]. После формирования ОС A СК с ОС B и C разных парных ключей (вероятность совпадения парных ключей (ПК) очень мала и стремится к нулю при неограниченном увеличении формируемого СК) ОС A формирует 2 закрытых (защищенных) канала, выбирает один из ПК за общий СК и передает его по закрытому каналу тому ОС, у которого другой ПК. Второй случай связан с формированием СК, когда ведется передача информации одновременно по обоим составляющим каналам ДШКБП. Для краткости модель, описывающую первый случай процесса формирования СК, назовем моделью формирования СК № 1 (МФШК-1), а модель, описывающую второй случай процесса формирования СК, назовем моделью формирования СК № 2 (МФШК-2).

В [1] показано, что в условиях одновременной передачи информации одновременно по обоим составляющим каналам ДШКБП, может быть затрачено меньше времени на передачу, чем при поочередной передаче информации по каждому составляющему каналу ДШКБП. Это определяет возможный выигрыш по времени формирования СК в МФШК-2 по сравнению с МФШК-1.

Одним из важнейших аспектов синтеза систем формирования СК является время формирования СК (или «информационная» скорость его формирования), т.к. это связано с информационными потерями, связанными с задержкой конфиденциальной информации при компрометации СК и необходимости временных затрат на установление криптосвязности на новом (не скомпрометированном) СК [19].

Показатель эффективности

Эффективность МФШК-1 и МФШК-2 предлагается в разрабатываемой модели оценки асимптотической эффективности открытого сетевого ключевого согласования оценивать показателем временной эффективности, т.е. исследовать соотношение временных показателей ПФСК с поочередной передачей информации, которая соответствует случаю разделения времени и ПФСК с одновременной передачей информации по ДШКБП. Считаем, что все процессы обработки информации у ОС не оказывают существенного влияния на время формирования СК (т.е. выполняются мгновенно) за исключением самого процесса передачи сообщений по каналам связи. Определим этот показатель и опишем модель оценивания этого показателя.

Пусть в модели МФШК-1 скорость формирования СК для двух ОС R_{11} [5, 7] с использованием канала от ОС A к ОС B равна

$$R_{11} = \frac{H(G^k)}{n_{11}}, \quad (1)$$

где $H(G^k)$ – информация ансамбля СК (сообщений), n_{11} – длина последовательности, передаваемой по каналу от ОС A к ОС B (первому составляющему каналу ДШКБП) [2, 17].

Пусть в модели МФШК-1 скорость формирования СК для двух ОС R_{12} с использованием канала от ОС A к ОС C равна

$$R_{12} = \frac{H(G^k)}{n_{12}}, \quad (2)$$

где n_{12} – длина последовательности передаваемой по каналу от ОС A к ОС C (второму составляющему каналу ДШКБП).

Пусть в модели МФШК-2 скорость формирования СК для трех сетевых ОС H_3 с использованием составляющих каналов ДШКБП от ОС A к ОС B и C равна

$$H_3 = \frac{H(G^k)}{n_3}, \quad (3)$$

где n_3 – длина последовательности передаваемой по составляющим каналам ДШКБП от ОС A к ОС B и C.

Пусть техническая скорость передачи информации [7, 16] в МФШК-1 и МФШК-2 одинакова и равна ν бит/с.

Пусть для обеих моделей заданы требования к формируемому СК [2], которые совпадают.

Тогда в рамках МФШК-1 для модели Чисара и Кернера можно сформировать СК, удовлетворяющий требованиям [2] с использованием дискретного канала без памяти от ОС A к ОС B, причем скорость формирования СК R_{11}, $R_{11} < C_{11}$, где C_{11} – значение ключевой пропускной способности двух ОС [10] для дискретного канала без памяти от ОС A к ОС B. Это можно сказать о процессе формирования СК с использованием дискретного канала без памяти от ОС A к ОС C, т.е. $R_{12} < C_{12}$, где C_{12} – значение ключевой пропускной способности двух ОС для дискретного канала без памяти от ОС A к ОС C. Для МФШК-2 в соответствии с теоремой о ключевой пропускной способности процесса формирования ключа для трех объектов связи доказанной в [2], можно сформировать СК, удовлетворяющий требованиям с использованием составляющих каналов связи ДШКБП от ОС A к ОС B и C, причем скорость формирования СК для трех сетевых ОС $H_3 < C_3$, где C_3 – значение сетевой ключевой пропускной способности трех сетевых ОС для ДШКБП [2] от ОС A к ОС B и C.

Определим T_1 – время формирования общего СК в МФШК-1.

$$T_1 = \nu n_{11} + \nu n_{12} + \nu H(G^k) = \nu H(G^k)\left(\frac{1}{R_{11}} + \frac{1}{R_{12}} + 1\right). \quad (4)$$

Подобным образом определим T_2 – время формирования общего СК в МФШК-2.

$$T_2 = \nu n_3 = \nu H(G^k)\frac{1}{H_3}. \quad (5)$$

Определение 1. Пусть ОС A, B и C для формирования общего СК используют модель МФШК-1 и МФШК-2. Пусть техническая скорость передачи информации в МФШК-1 и МФШК-2 одинакова и равна ν бит/с. *Временной эффективностью процесса формирования СК* для трех сетевых ОС χ называется коэффициент равный отношению времени формирования СК в МФШК-1 к времени формирования СК в МФШК-2, при котором обеспечивается выполнение заданных требований к формируемому СК [2]

$$\chi = \frac{T_1}{T_2} = \frac{H_3\,(R_{11}+R_{12}+R_{11}\,R_{12})}{R_{11}\,R_{12}}. \quad (6)$$

Найдем потенциально достижимый коэффициент выигрыша по времени формирования СК χ_0, если для формирования (передачи) СК используются коды неограниченной длины, т.е. в случае, если $n_{11} \to \infty$, $n_{12} \to \infty$ и $n_3 \to \infty$. В соответствии с (6), приведенными результатами в [5, 10] для модели Чисара и Кернера и с теоремой о ключевой пропускной способности процесса формирования ключа для трех объектов связи [2] и χ_0 вычисляется из формулы

$$\chi_0 = \frac{C_3\,(C_{11}+C_{12}+C_{11}\,C_{12})}{C_{11}\,C_{12}}, \quad (7)$$

где в рамках МФШК-1 C_{11} – значение ключевой пропускной способности двух ОС для дискретного канала без памяти от ОС A к ОС B, C_{12} – значение ключевой пропускной способности двух ОС для дискретного канала без памяти от ОС A к ОС C и в рамках МФШК-2 C_3 – значение ключевой пропускной способности трех сетевых ОС для ДШКБП от ОС A к ОС B и C.

Определение 2. Временная эффективность процесса формирования СК для трех сетевых ОС при использовании для формирования (передачи) СК кодов неограниченной длины называется *коэффициентом асимптотического выигрыша по времени формирования СК* для трех сетевых ОС, обозначается через χ_0 и определяется согласно (7).

Методика оценки асимптотической эффективности открытого сетевого ключевого согласования, т.е. оценки χ_0 сводится к определению ключевых пропускных способностей дискретных симметричных каналов без памяти для двух ОС в модели Чисара и Кернера и ключевой пропускной способности ПФСК с использованием ДШКБП для трех сетевых ОС. На первом шаге определяются ключевые пропускные способности дискретных симметричных каналов без памяти для двух ОС. Ключевая пропускная способность дискретного симметричного канала без памяти C_K определяется согласно выражению из [5]

$$C_K = C - C_w, \quad (8)$$

где C – пропускная способность дискретного симметричного канала без памяти между двумя ОС [8] и C_w – пропускная способность КП (дискретного симметричного канала без памяти между ОС A и нарушителем E).

На втором шаге методики находится значение ключевой пропускной способности процесса формирования ключа для трех объектов связи [2].

На завершающем шаге оценивается асимптотический выигрыш по времени формирования СК для трех сетевых ОС с использованием формулы (7).

Оценка коэффициента асимптотического выигрыша по времени формирования сетевого ключа

Из определения 2, оценки пропускной способности двоичного ДСК (ДСКБП) [16, 17] для общего двоичного алфавита ($t = 2$) вытекает следующее следствие.

Следствие. Пусть каналы связи МФШК-1 и МФШК-2 описываются с помощью моделей ДСКБП. Тогда коэффициент асимптотического выигрыша по времени формирования СК для трех сетевых ОС равен

$$\chi_0 = \frac{(h(p_w) - h(p))(2h(p_w) + h(p_w)^2 + h(p_y)h(p_m) - (h(p_y) + h(p_m))(1+ h(p_w)))}{(h(p_w)^2 + h(p_y)h(p_m) - h(p_w)(h(p_y) + h(p_m))}. \quad (9)$$

где $h(l) = -l\,logl - (1-l)\,log(1-l)$ – энтропийная функция ДСКБП [5, 9], p_w – вероятность ошибки в КП, который описывается моделью ДСКБП, вероятность p равна

$$p = p_y(1 - p_m) + (1 - p_y)p_m, \quad (10)$$

где p_y - вероятность ошибки в первом ДСКБП канале в МФШК-1 и первом составляющем ДСКБП канале двоичного широковещательного канала без памяти (ДвШКБП), p_w – вероятность ошибки во втором ДСКБП канале в МФШК-1 и втором составляющем ДСКБП канале ДвШКБП.

В таблице 1 приведены оценки значений коэффициента асимптотического выигрыша по времени формирования СК для трех сетевых ОС и фиксированной вероятности ошибки в КП $p_w = 0{,}3$ для интервала изменения вероятностей ошибок в составляющих ДСКБП от 0 до 0,3.

В таблице 2 приведены оценки значений коэффициента асимптотического выигрыша по времени формирования СК для трех сетевых ОС и фиксированной вероятности ошибки в КП $p_w = 0{,}1$ для интервала изменения вероятностей ошибок в составляющих ДСКБП от 0 до 0,3.

Анализ таблиц 1 и 2 показывает, что улучшение качества КП приводит к уменьшению области эффективного использования МФШК-2, где $\chi_0 > 1$. Оценка коэффициента ограничена значениями

$$0 \le \chi_0 < 3. \quad (11)$$

Таблица 1

Оценки значений χ_0 для общего двоичного алфавита при $p_w = 0{,}3$

		p_m						
		0	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
p_y	0	2.8813	2.2699	1.8801	1.5795	1.3402	1.1495	0
	0.05	2.2699	1.8684	1.5169	1.2119	0.925	0.5555	0
	0.1	1.8801	1.5169	1.1773	0.8611	0.5291	0	0
	0.15	1.5795	1.2119	0.8611	0.5203	0.1377	0	0
	0.2	1.3402	0.925	0.5291	0.1377	0	0	0
	0.25	1.1495	0.5555	0	0	0	0	0
	0.3	0	0	0	0	0	0	0

Таблица 2

Оценки значений χ_0 для общего двоичного алфавита при $p_w = 0{,}1$

		p_m		
		0	**0.05**	**0.1**
p_y	0	2.469	1.5719	0
	0.05	1.5719	0.1919	0
	0.45	0	0	0

Нижней границы, равной 0, коэффициент достигает, если $C_3 = 0$. Верхней границе необходимо уделить особое внимание. Для этого сравним предложенную в [1] модель передачи информации 1 (МПИ-1) с МФШК-1. В обеих моделях производится поочередная передача сообщений, однако во второй модели дополнительно присутствует нарушитель E. Задача нарушителя сводится к получению (формированию) общего СК для трех сетевых ОС. Это обуславливает определенные ограничения в МФШК-1 по сравнению с МПИ-1. При выполнении поочередной передачи сообщений в МПИ-1 ОС A может каждый раз передавать ОС B и C одно и тоже сообщение. При выполнении поочередной передачи сообщений в МФШК-1 ОС A не может делать этого за исключением одного случая. Рассмотрим, почему ОС A так нельзя делать. Если ОС A выполняет поочередную передачу одного и того же сообщения $\bar{g}$, где $\bar{g} \in G^k$, в МФШК-1 и после этого выбирает его в качестве общего СК, тогда нарушается требование по скорости получения информации о СК нарушителем E [2], т.к. после обеих передач нарушитель имеет в наличии 2 версии одного и того же сообщения $\bar{z}_1$ и $\bar{z}_2$, где $\bar{z}_1, \bar{z}_2 \in Z^n$. Тогда информация его $I(\bar{g};(\bar{z}_1, \bar{z}_2))$ после обеих передач увеличивается и становится больше, чем информация $I(\bar{g};\bar{z}_1)$ при первой передаче (или второй), что не допустимо согласно требования по скорости получения информации о СК нарушителем E [2]. Покажем это с использованием свойств средней взаимной информации [20]:

$$I(\bar{g};(\bar{z}_1, \bar{z}_2)) = I(\bar{g};\bar{z}_1) + I(\bar{g};\bar{z}_2/\bar{z}_1) \ge I(\bar{g};\bar{z}_1) \quad (12)$$

Для того, чтобы этого избежать, необходимо ОС А при выполнении второй передачи снова генерировать сообщение $\bar{g}'$, где $\bar{g}' \in G^k$ (которое является кодовым словом асимптотического кода [21]) и передавать его. В этом случае вероятность выполнения неравенства (12) значительно уменьшится. После этого ОС A выбирает сформированный СК одного из ОС (например, с ОС B) за общий

СК и передает его другому ОС (например, ОС C) по каналу, закрытому с помощью СК, сформированного с этим ОС. Ранее было сказано, что имеется одно исключение. Рассмотрим ситуацию с формированием (передачей) $\bar{g}'$, где $\bar{g}' \in G^k$, когда КП находится в состоянии «обрыва», т.е. $p_w = 0{,}5$. Тогда неравенство (12) превращается в равенство, причем $I(\bar{g};(\bar{z}_1, \bar{z}_2)) = 0$, т.к. $\bar{z}_1$ и $\bar{z}_2$ статистически не зависят от $\bar{g}$. И тогда можно предположить, что нарушителя нет и для формирования СК можно использовать модель МПИ-1 из [1]. В этом случае χ_0 определятся из выражения для коэффициента асимптотического выигрыша по времени передачи сообщения по ДвШКБП κ_0, который приведен в [1].

Верхняя граница для χ_0 при $p_w = 0{,}5$ могла бы равняться 3, если ОС формируют СК с использованием МФШК-1 для которой $p_w = 0{,}5$, $p_y = 0$ и $p_m = 0$. Однако, случай для $p_w = 0{,}5$ исследован выше, поэтому χ_0 менее 3.

Если $p_y = 0$ (или $p_m = 0$), тогда χ_0 больше 1. Это объясняется тем, что в (7) первый множитель в числителе и знаменатель равны и сокращаются, а второй множитель в числителе будет всегда больше 1.

Анализ остальных значений таблиц 1 и 2 показывает, что при достаточно малых значениях вероятностей ошибок в составляющих ДвШКБП $p_y << 0$ и $p_m << 0$ показатель χ_0 больше 1. Однако, с ухудшением качества составляющих ДвШКБП $p_y \longrightarrow 0{,}5$ и (или) $p_m \longrightarrow 0{,}5$ χ_0 становится менее 1 (или вообще равен нулю, т.к. $C_3 = 0$) и соответственно при этих сочетаниях p_y и p_m использование МФШК-2 становится не эффективным. Объяснить это можно следующим образом. Введем коэффициент $K2$ из (9), равный отношению второго множителя числителя к произведению $C_{11}\,C_{12}$ ключевых пропускных способностей составляющих каналов ДвШКБП, описываемых моделями ДСКБП

$$K2 = \frac{(2h(p_w) - h(p_y) - h(p_m))}{(h(p_w)\,h(p_w) - h(p_y) - h(p_m) + h(p_y)\,h(p_m))}. \quad (13)$$

Анализ (13) показывает, что $K2 > 1$, т.к. второе слагаемое в (13) будет всегда больше 1, когда $C_3 > 0$, т.к. представляет собой отношение суммы ключевых пропускных способностей составляющих каналов ДвШКБП к их произведению. Коэффициент $K2$ определен при большем числе сочетаний p_y и p_m, если $p_w \longrightarrow 0{,}5$, и возрастает при увеличении p_y и p_m. При фиксированном p_w и увеличении p_y и p_m C_3 уменьшается быстрее (энтропийная функция ДвШКБП возрастает быстрее), чем возрастает коэффициент $K2$. Это приводит к тому, что коэффициент χ_0 уменьшается, что уменьшает область временной эффективности формирования СК в МФШК-2.

Заключение

Подводя итоги, отметим следующее. В работе исследована асимптотическая эффективность открытого сетевого ключевого согласования. В ходе научного поиска введено понятие асимптотической эффективности процесса формирования СК для трех ОС, которое описывается коэффициентом асимптотического выигрыша по времени формирования СК. Предложены модель и методика оценки асимптотической эффективности открытого сетевого ключевого согласования где, определяется коэффициент асимптотического выигрыша по времени формирования общего ключа для трех сетевых ОС. Возможны два варианта формирования СК. В первом случае СК может формироваться при поочередной передаче информации по каждому составляющему каналу ДШКБП, который соответствует формированию СК в модели Чисара и Кернера [5, 10]. После формирования ОС разных СК, ОС А формирует 2 закрытых канала, выбирает один из ключей за общий СК и передает его по закрытому каналу тому ОС, у которого другой ключ. Второй случай связан с формированием СК, когда ведется передача информации одновременно по обоим составляющим каналам ДШКБП. Модель, описывающая первый случай, названа моделью формирования СК № 1 (МФШК-1), а модель, описывающую второй случай – моделью формирования СК № 2 (МФШК-2). Одним из важнейших аспектов синтеза систем формирования СК является время формирования СК [22, 23, 24], т.к. это связано с информационными потерями, связанными с задержкой конфиденциальной информации, необходимой для передачи в сети, при компрометации СК и возникновении временных затрат на установление криптосвязности на новом СК для продолжения закрытого информационного обмена. Поэтому для МФШК-1 и МФШК-2 введен коэффициент асимптотического выигрыша по времени формирования СК для трех сетевых ОС χ_0, равный отношению времени формирования СК в МФШК-1 к времени формирования СК в МФШК-2 при выполнении заданных требований к формируемому СК и неограниченном увеличении длины СК. Оценки значений χ_0 для двоичного ДШКБП показывают, что при достаточно малых значениях вероятностей ошибок в составляющих ДСКБП χ_0 больше 1, что предопределяет преимущественные условия использования МФШК-2. Улучшение качества КП приводит к уменьшению области эффективного использования МФШК-2. Коэффициент ограничен интервалом значений $0 \le \chi_0 < 3$. Таким образом, высокое качество составляющих каналов ДСКБП определяет предпочтительное использование МФШК-2 в режиме

одновременного формирования СК. Полученные результаты углубляют ранее описанные результаты оценок эффективности различных известных моделей открытого ключевого согласования: Вайнера [25], Чисара и Кернера [5, 10], Мауера [8, 14], квантового согласования ключей [26, 27] и могут быть полезны исследователям для анализа различных перспективных подсистем информационной безопасности телекоммуникационных систем, включающих подсистемы управления криптографическими ключами [5] и криптографические системы защиты информации [8], для оценки и поиска путей достижения потенциальных возможностей по уменьшению времени восстановления криптографической связности объектов связи после компрометации сетевого ключа нарушителем.

Полученные результаты углубляют и расширяют известные теоретико-информационные оценки эффективности различных моделей ключевого согласования.

Литература

1. *Синюк А. Д., Тарасов А. А., Остроумов О. А. Метод оценки временной эффективности передачи информации дискретного широковещательного канала связи // Телекоммуникации. 2021. № 7. С. 10–17. DOI: 10.31044/1684-2588-2021-0-7-10-17. EDN JMFKNS.*
2. *Синюк А. Д., Остроумов, О. А. Оценка ключевой пропускной способности сети связи // Вестник компьютерных и информационных технологий. 2020. Т. 17. № 11(197). С. 47–54. DOI: 10.14489/vkit.2020.11. Pp. 047–054.*
3. *Zhang Qikun, Li Yongjiao, Gan Yong, Zheng Chuanyang, Luo Xiangyang, Zheng Jun Group Key Agreement Protocol Based on Privacy Protection and Attribute Authentication // IEEE Access. Volume: 7. Page(s): 87085–87096. DOI: 10.1109/ACCESS.2019.2926404.*
4. *Pinar Sen, Sung Hoon Lim, Young-Han Kim On the Optimal Achievable Rates for Linear Computation With Random Homologous Codes // IEEE Transactions on Information Theory (Volume: 66), Issue: 10, October 2020) Page(s): 6200–6221 Date of Publication: 20 July 2020 DOI: 10.1109/TIT.2020.3010253*
5. *Hongchao Zhou, Abbas El Gamal Network Information Theoretic Security with Omnipresent Eavesdropping // IEEE Transactions on Information Theory. Volume: 67. Issue: 12. December 2021. Page(s): 8280–8299. DOI: 10.1109/TIT.2021.3116962.*
6. *Onur Günlü, Rafael F. Schaefer, Holger Boche, H. Vincent Poor Secure and Private Distributed Source Coding With Private Keys and Decoder Side Information // IEEE Transactions on Information Forensics and Security (Volume: 18) Page(s): 3803–3816 Date of Publication: 14 June 2023. DOI: 10.1109/TIFS.2023.3286285*
7. *Tetsunao Matsuta; Tomohiko Uyematsu Coding Theorems for Asynchronous Slepian-Wolf Coding Systems // IEEE Transactions on Information Theory (Volume: 66), Issue: 8, August 2020), Page(s): 4774–4795, Date of Publication: 18 February 2020, ISSN Information: Print ISSN: 0018-9448 Electronic ISSN: 1557-9654, DOI: 10.1109/TIT.2020.2974736*
8. *Matthieu Bloch, Onur Günlü, Aylin Yener, Frédérique Oggier, H. Vincent Poor, Lalitha Sankar, Rafael F. Schaefer An Overview of Information-Theoretic Security and Privacy: Metrics, Limits and Applications // IEEE Journal on Selected Areas in Information Theory. Volume: 2. Issue: 1. March 2021. Page(s): 5–22. DOI: 10.1109/JSAIT.2021.3062755.*
9. *Остроумов О. А., Синюк А. Д. Пропускная способность широковещательного канала связи // Вестник компьютерных и информационных технологий. 2019. № 9 (183). С. 33–42. DOI: 10.14489/vkit.2019.09.pp.033-042.*
10. *Cheuk Ting Li; Venkat Anantharam One-Shot Variable-Length Secret Key Agreement Approaching Mutual Information // IEEE Transactions on Information Theory (Volume: 67), Issue: 8, August 2021) Page(s): 5509–5525 at of Publication: 09 June 2021 DOI: 10.1109/TIT.2021.3087963*
11. *Zhang Qikun, Li Yongjiao, Gan Yong, Zheng Chuanyang, Luo Xiangyang, Zheng Jun Group Key Agreement Protocol Based on Privacy Protection and Attribute Authentication // IEEE Access. Volume: 7. Page(s): 87085–87096. Date of Publication: 02 July 2019 Electronic ISSN: 2169-3536 INSPEC Accession Number: 18826825. DOI: 10.1109/ACCESS.2019.2926404.*
12. *Vamoua Yachongka, Hideki Yagi, Hideki Ochiai Key Agreement Using Physical Identifiers for Degraded and Less Noisy Authentication Channels // IEEE Transactions on Information Forensics and Security (Volume: 18) Page(s): 5316 – 5331, Date of Publication: 23 August 2023 DOI: 10.1109/TIFS.2023.3307976*
13. *Onur Günlü;Rafael F. Schaefer Controllable Key Agreement With Correlated Noise // IEEE Journal on Selected Areas in Information Theory (Volume: 2, Issue: 1, March 2021) Page(s): 82–94 Date of Publication: 25 January 2021 Electronic ISSN: 2641-8770 DOI: 10.1109/JSAIT.2021.3054035*
14. *Mohamed Nafea, Aylin Yener Generalizing Multiple Access Wiretap and Wiretap II Channel Models: Achievable Rates and Cost of Strong Secrecy // IEEE Transactions on Information Theory. Volume: 65. Issue: 8. August 2019. Page(s): 5125 – 5143. DOI: 10.1109/TIT.2019.2908832.*
15. *Остроумов О. А., Синюк А. Д. Информационная скорость формирования сетевого ключа по открытым виртуальным каналам связи // Вопросы кибербезопасности. 2023. № 3(55). с. 78–89. DOI: 10.21681/2311-3456-2023-3-78-89.*
16. *Anuran Makur Coding Theorems for Noisy Permutation Channels // IEEE Transactions on Information Theory (Volume: 66, Issue: 11, November 2020) Page(s): 6723–6748 Date of Publication: 16 July 2020. DOI: 10.1109/TIT.2020.3009468*
17. *Haoheng Yuan, Yanghe Feng, Chuanchuan Yang, Zhuojun Zhuang, Bin Dai Two-User Gaussian Broadcast Wiretap Channel With Common Message and Feedback: Revisit // IEEE Transactions on Information Forensics and Security (Volume: 19) Page(s): 178–193 Date of Publication: 25 September 2023. DOI: 0.1109/TIFS.2023.3318948*
18. *Meryem Benammar, Pablo Piantanida, Shlomo Shamai on the Compound Broadcast Channel: Multiple Description Coding and Interference Decoding // IEEE Transactions on Information Theory (Volume: 66). Issue: 1, January 2020) Page(s): 38–64 Date of Publication: 23 September 2019. DOI: 10.1109/TIT.2019.2942615*
19. *Alejandro Cohen, Rafael G. L. D'Oliveira, Salman Salamatian, Muriel Médard Network Coding-Based Post-Quantum Cryptography // IEEE Journal on Selected Areas in Information Theory (Volume: 2, Issue: 1, March 2021) Page(s): 49 – 64 Date of Publication: 26 January 2021 Electronic ISSN: 2641-8770. DOI: 10.1109/JSAIT.2021.3054598*

20. *Cheuk Ting Li, Venkat Anantharam One-Shot Variable-Length Secret Key Agreement Approaching Mutual Information // IEEE Transactions on Information Theory. Volume: 67. Issue: 8. August 2021. Page(s): 5509–5525. DOI: 10.1109/TIT.2021.3087963.*
21. *Vidhi Rana, Rémi A. Chou, Hyuck M. Kwon Information-Theoretic Secret Sharing From Correlated Gaussian Random Variables and Public Communication // IEEE Transactions on Information Theory (Volume: 68), Issue: 1, January 2022) Page(s): 549–559 Date of Publication: 27 October 2021. DOI: 0.1109/TIT.2021.3122808*
22. *Starostin V., Korzhik V., Kabardov M., Gerasimovich A., Yakovlev V., Morales-Luna G Key generation protocol executing through non-reciprocal fading channels // International Journal of Computer Science and Applications. 2019. Т. 16. № 1. С. 1–16.*
23. *Синюк А. Д., Тарасов А. А., Остроумов О. А. Теоретико-информационное представление виртуализации сетевого канала перехвата // Информатика и автоматизация. 2023. Т. 2. № 4. с. 721–744. DOI: 10.15622/ia.22.4.1.*
24. *Синюк А. Д., Остроумов О. А. Теорема о ключевой пропускной способности сети связи // Информационно-управляющие системы. 2018. № 5(96). с. 79–87. DOI: 10.31799/1684-8853-2018-5-79-87.*
25. *Amin Gohari, Onur Günlü, Gerhard Kramer Coding for Positive Rate in the Source Model Key Agreement Problem // IEEE Transactions on Information Theory. Volume: 66. Issue: 10. October 2020. Page(s): 6303–6323. DOI: 10.1109/TIT.2020.2990750.*
26. *Ignazio Pedone, Andrea Atzeni, Daniele Canavese, Antonio Lioy Toward a Complete Software Stack to Integrate Quantum Key Distribution in a Cloud Environment // IEEE Access (Volume: 9) Page(s): 115270–115291 Date of Publication: 03 August 2021 Electronic ISSN: 2169-3536 DOI: 10.1109/ACCESS.2021.3102313*
27. *Yi Luo; Hao-Kun Mao; Qiong Li; Nan Chen An Information-Theoretic Secure Group Authentication Scheme for Quantum Key Distribution Networks // IEEE Transactions on Communications (Volume: 71), Issue: 9, September 2023) Page(s): 5420–5431. Date of Publication: 29 May 2023. DOI: 10.1109/TCOMM.2023.3280561*

О МОДЕЛЯХ ПОСТРОЕНИЯ ГРАФА ВЗАИМОДЕЙСТВУЮЩИХ ОБЪЕКТОВ В СЕТИ TELEGRAM-КАНАЛОВ

Попов В. А.[1], Чеповский А. А.[2]

DOI: 10.21681/2311-3456-2024-3-105-112

Цель исследования: *сравнение широкого набора различных моделей построения графов взаимодействующих объектов в сети публичных Telegram-каналов с целью выявления среди них наиболее подходящих, при которых полученный граф наиболее близок к безмасштабным сетям.*

Метод исследования: *для построенных взвешенных графов в рамках каждой из рассматриваемых моделей находятся степенные законы, наиболее приближающие эмпирические распределения полученных весов вершин, после чего оценивается качество полученного приближения.*

Полученный результат: *в статье представлены модели построения графов, характеризующих информационное воздействие в сети Telegram-каналов. В данной работе представлены результаты исследования 180 случаев – для 12 моделей проведены исследования на 15 наборах данных. В рамках этих исследований найдены параметры степенных законов, приближающих эмпирические данные. Показано, у каких из моделей эти параметры оказываются не свойственными для безмасштабных сетей. С помощью критерия Колмогорова проверены гипотезы о характере распределения у моделей. Приведены иллюстрации, наглядно показывающие результаты исследования. Показано, какая из моделей лучше всего подходит для формирования графов взаимодействующих объектов в сети Telegram-каналов. Такие графы могут быть впоследствии проанализированы с целью выделения ключевых вершин.*

Научная новизна: *предложены модели для представления имевшего место взаимодействия объектов сети Telegram-каналов в виде взвешенных графов. Исследовано распределение весов вершин у полученных графов взаимодействующих объектов. Изучение этого важного свойства для взвешенных графов, полученных при импорте данных из реальных сетей, дало важный теоретический и практический результат. Выявлено, что UMR-модель построения таких графов, обладает свойством, характерным для безмасштабных сетей.*

Ключевые слова: *безмасштабные сети, модель информационного воздействия, выделение сообществ, анализ социальных сетей, критерий согласия Колмогорова, степенной закон распределения, вес вершин.*

ABOUT MODELS TO CONSTRUCT A GRAPH OF INTERACTING OBJECTS IN A NETWORK OF TELEGRAM CHANNELS

Popov V. A.[3], Chepovskiy A. A.[4]

The purpose of the study: *comparison of a wide range of different models to construct graphs of interacting objects in a public Telegram channels network in order to identify among them the most suitable ones, in which the resulting graph is closest to scale-free networks.*

Method: *for the constructed weighted graphs, within the framework of each of the models under consideration, power laws are found that most closely approximate the empirical distributions of the obtained vertices weights, after which the quality of the resulting approximation is assessed.*

1 Попов Владимир Александрович, аспирант Департамента прикладной математики МИЭМ НИУ ВШЭ, Москва, Россия. E-mail: vapopov@hse.ru
2 Чеповский Александр Андреевич, кандидат физико-математических наук., доцент, Департамент прикладной математики МИЭМ НИУ ВШЭ, Москва, Россия. E-mail: aachepovsky@hse.ru

3 Vladimir A. Popov, Ph.D. student, School of Applied Mathematics, HSE MIEM, Moscow, Russia. E-mail: vapopov@hse.ru
4 Alexander A. Chepovskiy, Ph.D. in Physics and Mathematics, Associate Professor, Department of Applied Mathematics, Moscow Institute of Economics, National Research University Higher School of Economics, Moscow, Russia. E-mail: aachepovsky@hse.ru

Results: *the article presents models to construct graphs that characterize the information impact in the Telegram channels network. This paper presents the results of a study of 180 cases – studies were conducted for 12 models on 15 data sets. As part of these studies, parameters of power laws that approximate empirical data were found. It is shown which of the models have these parameters that are not characteristic of scale-free networks. Using the Kolmogorov criterion, hypotheses about the nature of the distribution of the models were tested. Illustrations are provided to clearly show the results of the study. It is shown which of the models is best suited to construct graphs of interacting objects in a network of Telegram channels. Such graphs can subsequently be analyzed to identify key vertices.*

Scientific novelty: *models are proposed to represent the interaction of objects in the Telegram channel network in the form of weighted graphs. The distribution of vertex weights in the resulting graphs of interacting objects has been researched. Studying this important property for weighted graphs obtained by importing data from real networks has yielded important theoretical and practical results. It was revealed that the UMR-model to construct such graphs has a property characteristic of scale-free networks.*

Keywords: *scale-free networks, model of information impact, community detection, analysis of social networks, Kolmogorov goodness-of-fit test, power law distribution, vertex weight.*

1. Введение

В науке много лет известно понятие сложных сетей (complex networks) – биологических, технологических, телекоммуникационных, социальных сетей, содержащих большое число объектов и относительно малое число связей между ними. Такие сети, возникающие на практике, относятся к так называемым безмасштабным сетям (free-scale networks)[5]. Анализ графов, характеризующих безмасштабные сети является предметом подробных исследований многих авторов последних двух десятилетий [1, 2], построены различные алгоритмы, позволяющие выделять на графе подграфы, именуемые неявными сообществами, а также ключевые вершины графа, используя специализированные центральности [3, 4].

В контексте практических исследований для телекоммуникационных сетей важным является импорт реальных данных и построение на их основе графов взаимодействующих объектов – графов, характеризующих наличие коммуникации между объектами исходной сети или степень ее интенсивности, а также метаданные о содержании этого взаимодействия, включая текстовую информацию. Одной из актуальных для анализа данных является сеть Telegram-каналов. Помимо обмена личными сообщениями между пользователями в Telegram реализована функциональность для организации публичных Telegram-каналов, представляющих собой информационно-новостные ленты сообщений. Многие СМИ, информационные сообщества, блогеры имеют свои Telegram-каналы и регулярно публикуют в них контент, а пользователи Telegram могут подписаться на данные каналы и получать информацию в виде сообщений от имени канала.

При этом Telegram предоставляет широкий спектр инструментов для ведения каналов. Помимо уникального текстового и фото-видео контента, администратор канала может опубликовать пост другого канала в своем (сделать репост), процитировать или упомянуть другие Telegram-каналы, сделать ссылку на внешний адрес в сети интернет. Использование данных возможностей создает связи между Telegram-каналами, что позволяет рассматривать их сеть для импорта данных и построения взвешенного графа взаимодействующих объектов. Так, авторами в предыдущих работах была представлена (U,M,R)-модель [5] и алгоритм для последующего выделения на построенном с ее помощью графе неявных сообществ [6].

В практическом плане одной из актуальных задач является выбор модели при построении графа, ибо в зависимости от этого выбора дальнейший анализ, основанный в том числе на весе ребер и вершин, будет давать разный результат. Поэтому важно выбрать модель, при которой построенный граф обладает свойствами безмасштабных сетей.

В данной работе в первой части описывается общая модель построения графов взаимодействующих объектов и 12 моделей, при которых по-разному определяется вес на ребрах графа. После этого показано, как по набору весов вершин графа определить параметры степенного распределения, приближающего данный набор значений. На основе полученного приближения сравниваются рассматриваемые модели. Далее демонстрируются примеры приближения эмпирических данных степенными функциями, для них строятся и сравниваются функции распределения и функции плотности вероятности.

5 Fortunato S. Community Detection in Graphs // Physics Reports. — 2010. — 486(3). — P. 75–174.
Евин И. А. Введение в теорию сложных сетей // Компьютерные исследования и моделирование. — 2010. — 2(2). — С. 121–141.

2. Модели построения графа и наборы данных

В данной работе рассматриваются публичные каналы в мессенджере Telegram, для которых можно выделить следующие ключевые факторы взаимодействия: репосты между каналами, упоминания одного канала другим и наличие общих внешних URL в постах двух каналов. Для построения таких моделей изначально необходимо импортировать данные из мессенджера за выбранный период времени, а именно данные о постах, интересующих Telegram-каналов. С этой целью было разработано программное обеспечение, использующее официальный API Telegram, способное импортировать информацию о каналах в данные специального AVS-формата [6, 7], содержащие все необходимые составляющие для дальнейшего анализа и выявления взаимодействий между каналами. Для построения графа взаимодействующих объектов первично формируется полный граф G(V,E), содержащий все импортированные вершины, соответствующие каналам, и все возможные ребра между ними. Далее осуществляется переход к взвешенному графу $G(V,\tilde{E})$, где на ребрах задан вес, определенный исходя из выявленного взаимодействия между соответствующими вершинами. В случае нулевого веса ребра удаляются. При этом исходный импорт данных и построение множества V устроены таким образом, что граф $G(V,\tilde{E})$ представляет из себя одну компоненту связности.

Зададим на исходном множестве ребер E весовую функцию w, зависящую от выявленных факторов взаимодействия $\delta^U_{e_{AB}}$, $\delta^M_{e_{AB}}$, $\delta^R_{e_{AB}}$:

$$w(e_{AB}) = \mathrm{F}(\delta^U_{e_{AB}},\delta^M_{e_{AB}},\delta^R_{e_{AB}}), \quad (1)$$

где $\delta^U_{e_{AB}}$ – количество общих уникальных внешних ссылок (URL) в постах у каналов A и B за выбранный период; $\delta^M_{e_{AB}}$ – количество постов, где в тексте канал A упомянул канал B плюс количество постов, где B упомянул A за выбранный период (для каждого поста смотрятся уникальные упоминания, то есть если в одном посте канал A упомянул несколько раз канал B, то это упоминание все равно учитывается единожды в данном коэффициенте; $\delta^R_{e_{AB}}$ – количество репостов каналом A сообщений канала B, плюс количество репостов каналом B сообщений канала A за выбранный период; F – функция, зависящая от $\delta^U_{e_{AB}}$, $\delta^M_{e_{AB}}$, $\delta^R_{e_{AB}}$, возвращает неотрицательные значения.

Различные модели определяются тем, какую функцию F выбрать при определении веса ребра между вершинами. Далее рассмотрим 12 моделей, соответствующих разным функциям F. Первые две из них являются (U,M,R)-моделью [5] с различающимися значениями параметров U,M и R и представляют из себя линейные функции:

$$F_{1,1} = 1 \cdot \delta^U_{e_{AB}} + 2 \cdot \delta^M_{e_{AB}} + 3 \cdot \delta^R_{e_{AB}} \quad (2)$$

$$F_{1,2} = 1 \cdot \delta^U_{e_{AB}} + 1 \cdot \delta^M_{e_{AB}} + 1 \cdot \delta^R_{e_{AB}} \quad (3)$$

Формулы (2) и (3) означают, что для $F_{1,1}$ в базовой (U,M,R)-модели взяты значения $U = 1$, $M = 2$, $R = 3$. А для $F_{1,2}$ взяты значения $U = M = R = 1$. Помимо них рассмотрим более сложные конструкции, не подпадающие под (U,M,R)-модель, но основанные на тех же факторах взаимодействия как аргументах для функций F.

Для второй группы рассматриваемых функций используем логарифмическую составляющую:

$$F_{2,1} = 1 \cdot \ln(\delta^U_{e_{AB}} + 1) + 1 \cdot \ln(\delta^M_{e_{AB}} + 1) + 1 \cdot \ln(\delta^R_{e_{AB}} + 1) \quad (4)$$

$$F_{2,2} = 1 \cdot \ln(\delta^U_{e_{AB}} + 1) + 2 \cdot \ln(\delta^M_{e_{AB}} + 1) + 3 \cdot \ln(\delta^R_{e_{AB}} + 1) \quad (5)$$

Добавим в анализируемый набор варианты для F, у которых есть компоненты из показательных функций. В указанных ниже функциях под $max\ \delta^U_e$, $max\ \delta^M_e$, $max\ \delta^R_e$ мы будем понимать максимальные по всему графу $G(V,E)$ значения для δ^U_e, δ^M_e, δ^R_e соответственно:

$$F_{3,1} = 2^{\frac{\delta^U_{e_{AB}}}{max\ \delta^U_e}} + 2^{\frac{\delta^M_{e_{AB}}}{max\ \delta^M_e}} + 2^{\frac{\delta^R_{e_{AB}}}{max\ \delta^R_e}} - 3 \quad (6)$$

$$F_{3,2} = 2^{\frac{\delta^U_{e_{AB}}}{max\ \delta^U_e}} + 2^{2\frac{\delta^M_{e_{AB}}}{max\ \delta^M_e}} + 2^{3\frac{\delta^R_{e_{AB}}}{max\ \delta^R_e}} - 3 \quad (7)$$

$$F_{3,3} = 2^{\frac{\delta^U_{e_{AB}}}{max\ \delta^U_e}} + 2 \cdot 2^{\frac{\delta^M_{e_{AB}}}{max\ \delta^M_e}} + 3 \cdot 2^{\frac{\delta^R_{e_{AB}}}{max\ \delta^R_e}} - 6 \quad (8)$$

И рассмотрим еще несколько вариантов для функции F, скомбинированных из предыдущих:

$$F_{4,1} = \delta^U_{e_{AB}} + \ln(\delta^M_{e_{AB}} + 1) \cdot \ln(\delta^R_{e_{AB}} + 1) \quad (9)$$

$$F_{4,2} = (\delta^U_{e_{AB}} + 1) \cdot \ln(\delta^M_{e_{AB}} + 1) + \ln(\delta^R_{e_{AB}} + 1) \quad (10)$$

$$F_{4,3} = (\delta^U_{e_{AB}} + 1) \cdot \ln(\delta^R_{e_{AB}} + 1) + \ln(\delta^M_{e_{AB}} + 1) \quad (11)$$

$$F_{4,4} = (\delta^U_{e_{AB}} + 1) \cdot (\ln(\delta^U_{e_{AB}} + 1) + \ln(\delta^R_{e_{AB}} + 1)) \quad (12)$$

$$F_{4,5} = (\delta^U_{e_{AB}} + 1) \cdot (2 \cdot \ln(\delta^U_{e_{AB}} + 1) + 3 \cdot \ln(\delta^R_{e_{AB}} + 1)) \quad (13)$$

Для сравнения моделей, полученных на основе описанных выше весовых функций, были скачаны данные по 15 сетям. Импорт данных для каждой из них проводился от некоторых стартовых вершин на наперед заданную глубину по фиксированному для каждого из случаев временному периоду. Как и было описано в начале раздела, далее на основе этих импортированных данных были сформированы графы $G_i(V,E)$. До применения функции F и перехода к графам $G_i(V,\tilde{E})$ можно посмотреть на характеристики скачанных подсетей с учетом выявленных факторов взаимодействия и ребер, вес на которых будет ненулевым. Они едины для всех весовых функций, применяемых к $G_i(V,E)$ и зависят именно от отсутствия имевшего место взаимодействия. Так же можно подсчитать для таких графов диаметр и классический коэффициент кластеризации, они не будут зависеть от весов ребер, а только от наличия ненулевых ребер. Такие данные приведены в табл. 1.

Таблица 1.

Характеристики импортированных подсетей

Граф	N – количество вершин в графе $G_i(V,\tilde{E})$	M – количество ненулевых ребер в графе $G_i(V,\tilde{E})$	Диаметр графа	Коэффициент кластеризации графа
G_1	179	723	6	0,29
G_2	185	919	4	0,31
G_3	302	2641	4	0,31
G_4	437	4892	4	0,30
G_5	451	3580	6	0,27
G_6	464	1921	6	0,24
G_7	590	4352	9	0,22
G_8	600	18009	4	0,32
G_9	619	2973	7	0,22
G_{10}	625	6137	6	0,20
G_{11}	773	6611	6	0,18
G_{12}	1252	17841	4	0,28
G_{13}	1432	44595	4	0,24
G_{14}	1592	34877	4	0,19
G_{15}	3736	93215	6	0,21

Далее на ребрах этих графов применяются весовые функции F, описанные ранее. За счет чего получаем графы $G_i(V,\tilde{E})$, с которыми и будут проведены дальнейшие действия. Фактически для каждой из 15 сетей мы получили по 12 различных взвешенных графов и общий набор данных составил 180 взвешенных графов. Вес вершины определяется как сумма весов инцидентных ей ребер.

3. Распределение весов вершин графов

В работах, посвященным безмасштабным сетям[6], полученным на основе взаимодействия социальных объектов, показывается, что распределение весов вершин подчиняется степенному закону с параметром степени $\alpha \in [2; 3]$:

$$f(x) = P(X = x) = Cx^{-\alpha} \quad (14)$$

Это свойство используется в методах генерации случайных сетей при моделировании сложных сетей [8, 9].

На рассматриваемом наборе графов сравним эмпирические значения весов вершин и получаемые распределения для определения лучшей весовой функции. Под весом вершины понимаем сумму весов всех ребер, инцидентных этой вершине. Для каждого из 180 рассматриваемых графов будем приближать эмпирические распределения весов вершин $\{x_i\}_{i=1}^{N}$ функциями со степенным законом, находить оптимальный параметр степени α и считать соответствующую ошибку – насколько хорошо распределения приближают эмпирические данные. Для нахождения параметров степенных распределений, которые наиболее точно описывают полученные эмпирические распределения весов вершин, воспользуемся методом максимального правдоподобия и критерием согласия Колмогорова[7].

Степенной закон, указанный в формуле (14), у безмасштабных сетей имеет место для «хвоста» распределения, начиная с некоторого значения x_{min}. Будем приближать не все эмпирическое распределение, а только часть, для которой выполняется условие:

$$x_i > x_{min} \quad (15)$$

Это означает, что реально рассматривается не вся выборка $\{x_i\}_{i=1}^{N}$, а только ее часть $\{x_i\}_{i=1}^{k}$, которая удовлетворяет условию (15). При таком подходе возможна ситуация, что эмпирические данные приближаются степенной функцией наилучшим образом при слишком высоком значении x_{min}. А это означает, что число вершин k, для которых будет рассматриваться выборка $\{x_i\}_{i=1}^{k}$ мало. Поэтому введем еще одно ограничение, которое позволяет рассматривать содержательно существенное число вершин. А именно, будем рассматривать такие x_{min}, что количество вершин k удовлетворяет условию:

$$k > 0{,}2 \cdot N \quad (16)$$

Теперь перейдем к рассмотрению $\{x_i\}_{i=1}^{k}$. Для начала с помощью метода максимального правдоподобия находится показатель степени распределения $\hat{\alpha}$, считая, что значение x_{min} известно. Данный метод доказуемо дает точные оценки параметров моделей для выборок большого размера[8]. После этого, минимизируя расстояние Колмогорова-Смирнова,

6 Newman M. E. J. Networks: An Introduction. – Oxford University Press, 2010. – 784 p.
G. Caldarelli, Scale-Free Networks. Oxford University Press, Oxford (2007).

7 Clauset A, Shalizi CR, Newman MEJ Power-law distributions in empirical data. SIAM Review 51: 661–703 (2009).

8 Wasserman, L., All of Statistics: A Concise Course in Statistical Inference (Springer-Verlag, Berlin), 2003.
Barndorff-Nielsen, O. E., and D. R. Cox, Inference and Asymptotics (Chapman and Hall, London), 1995.

Таблица 2

Полученные значения $\hat{\alpha}$ для x_{min} с учетом ограничения (16)

	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
$F_{1,1}$	2,15	2,07	2,36	2,03	2,00	2,06	2,25	2,31	2,22	2,31	2,09	2,35	2,01	1,95	2,00
$F_{1,2}$	1,81	2,05	2,48	2,20	2,08	2,16	2,19	2,34	2,22	2,35	1,83	2,26	2,06	1,99	2,13
$F_{2,1}$	2,38	2,11	2,96	2,82	2,26	2,26	2,83	2,46	2,03	2,25	2,52	1,96	2,17	2,45	2,41
$F_{2,2}$	2,92	2,32	2,87	2,65	2,47	2,43	2,74	2,75	2,64	2,48	2,47	2,64	2,24	2,41	2,33
$F_{3,1}$	1,82	2,37	2,69	1,72	1,94	2,31	2,21	2,25	2,20	2,11	2,12	2,04	2,05	2,02	2,26
$F_{3,2}$	1,87	2,11	2,33	1,84	2,06	2,14	2,24	2,22	1,72	2,30	2,08	2,12	2,20	2,09	2,02
$F_{3,3}$	2,15	2,20	2,59	1,87	2,07	2,14	2,29	1,88	1,70	2,27	2,13	2,13	2,21	1,99	2,03
$F_{4,1}$	2,16	2,62	2,79	1,79	1,97	2,40	2,68	2,53	2,44	2,29	2,12	2,10	2,20	2,14	2,24
$F_{4,2}$	1,85	2,07	2,25	2,11	2,09	2,14	2,16	2,34	2,31	2,21	2,25	2,26	2,03	2,13	2,02
$F_{4,3}$	1,96	2,07	2,31	2,21	2,13	1,95	2,09	2,51	2,48	2,38	2,25	2,42	2,06	2,24	1,98
$F_{4,4}$	1,73	1,91	2,04	2,02	1,98	1,99	2,70	2,30	2,27	2,20	2,16	2,22	2,02	2,21	1,92
$F_{4,5}$	2,06	1,92	1,96	2,04	2,02	1,91	2,68	2,36	2,25	2,41	2,18	2,23	2,01	2,22	1,90

выбирается значение x_{min}. С учетом условия (16) это будет не абсолютный минимум x_{min} для рассматриваемого графа, а локальный в этом интервале.

Возьмем графы, полученные для функций (2)–(13), и посмотрим на значение расстояния Колмогорова-Смирнова для оценки разницы между эмпирической функцией распределения вершин и найденной степенной функцией. Для каждого графа $G_i(V,\tilde{E})$ и каждой из функций подсчитана статистика Колмогорова [10]:

$$D_k = \sup_{x \geq x_{min}} |f(x) - \hat{f}_k(x)| \qquad (17)$$

где $f(x)$ – найденная степенная функция распределения, а $\hat{f}_k(x)$ – эмпирическая функция распределения, k – размер соответствующей выборки (число рассматриваемых вершин).

Минимизируя D_k и находя так локальный минимум x_{min}, находится и значение $\hat{\alpha}$. Полученные значения $\hat{\alpha}$ для разных весовых функций на исследуемом наборе графов с учетом описанных ранее условий представлены в таблице 2. Как видно, не во всех случаях выполнено $\hat{\alpha} \in [2; 3]$.

Используем критерий согласия Колмогорова для проверки на уровне значимости в 10% гипотезы о том, что эмпирически полученный для каждой из весовых функций набор весов вершин соответствует степенному закону распределения. Для этого проверим, что выполняется следующее условие:

$$\sqrt{k}\, D_k < z_{0,9}, \qquad (18)$$

где $z_{0,9} = 1{,}22$ – квантиль распределения Колмогорова для уровня значимости 10%.

Таблица 3

Результаты проверки критерия Колмогорова

	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
$F_{1,1}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ЛОЖЬ
$F_{1,2}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ЛОЖЬ
$F_{2,1}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ЛОЖЬ
$F_{2,2}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ЛОЖЬ
$F_{3,1}$	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ЛОЖЬ
$F_{3,2}$	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ
$F_{3,3}$	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ЛОЖЬ
$F_{4,1}$	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ
$F_{4,2}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ
$F_{4,3}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ЛОЖЬ	ИСТИНА	ЛОЖЬ	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ
$F_{4,4}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ
$F_{4,5}$	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ИСТИНА	ЛОЖЬ	ИСТИНА	ЛОЖЬ

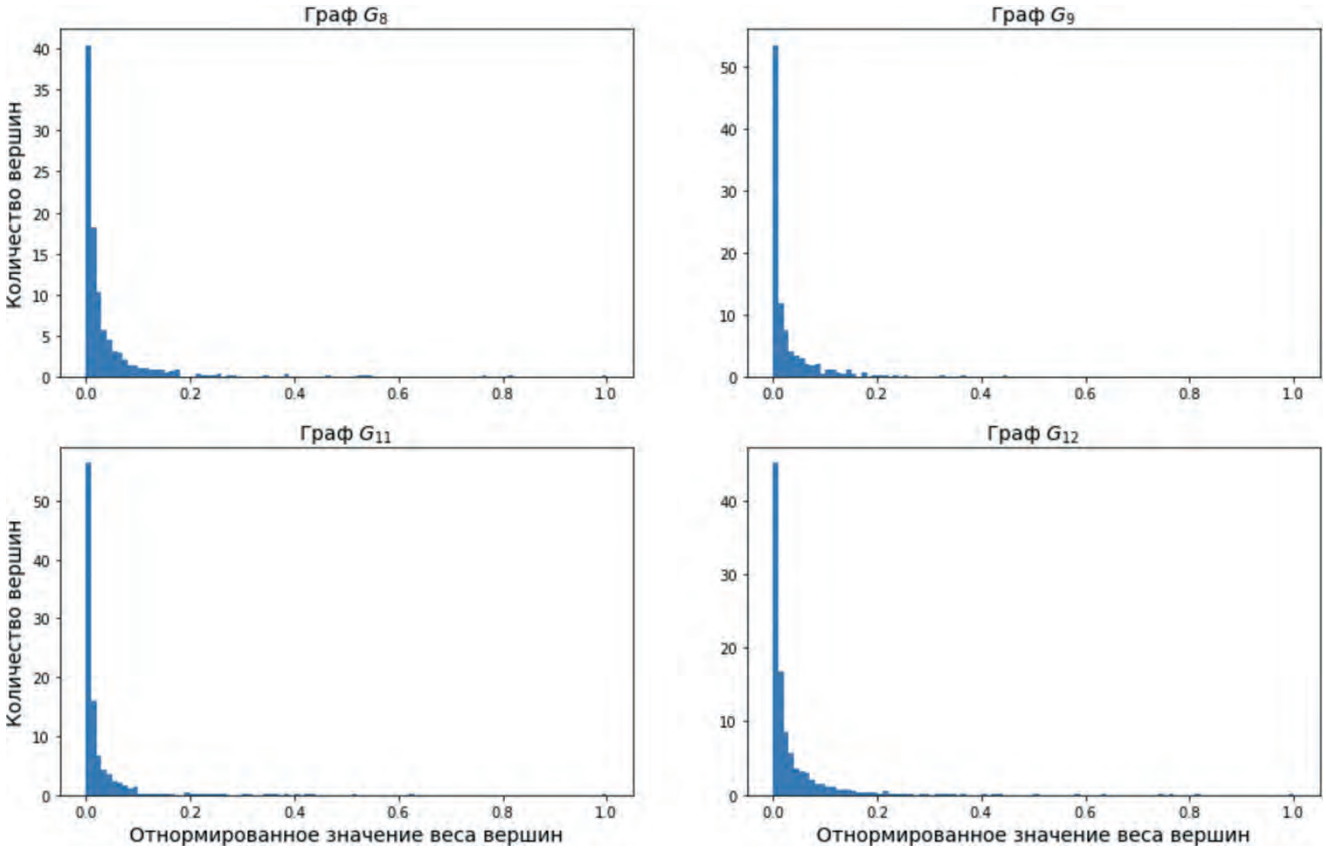


Рис 1. Гистограммы распределения весов вершин на графах $G(V,\tilde{E})$ для G_8, G_9, G_{11}, G_{12} при весовой функции $F_{1,1}$.

В таблице 3 представлены результаты проверки критерия. Из таблицы видно, что получаемый для весовой функции $F_{1,1}$ набор весов вершин соответствует степенному распределению с найденным параметром $\hat{\alpha}$. Это подтверждает целесообразность использования $F_{1,1}$ при построении взвешенных графов взаимодействующих объектов, обладающих свойствами безмасштабных сетей.

4. Примеры приближения эмпирических данных степенной функцией

Убедимся еще и визуально, что полученные распределения весов вершин графов подчиняются степенному закону. Для этого на примере функции $F_{1,1}$ и графов G_8, G_9, G_{11}, G_{12} из анализируемого набора построим гистограммы, представляющие собой частотные распределения (рис. 1). Визуально распределения весов вершин для функции $F_{1,1}$ удовлетворяет ожиданиям.

Далее для рассматриваемых четырех эмпирических распределений весов вершин найдем функции плотности вероятности для выборок $\{x_i\}_{i=1}^{k}$, удовлетворяющие условиям (15) и (16), и построим их на логарифмических осях (рис. 2). Также на этих графиках построим соответствующие плотности вероятности теоретических степенных распределений, параметры которых найдены в разделе 3 при помощи метода максимального правдоподобия:

- Для графа G_8 – степенное распределение с параметром $\alpha = 2.31$
- Для графа G_9 – степенное распределение с параметром $\alpha = 2.22$
- Для графа G_{11} – степенное распределение с параметром $\alpha = 2.09$
- Для графа G_{12} – степенное распределение с параметром $\alpha = 2.35$

Графиком плотности вероятности степенного распределения, построенном на логарифмических осях, является прямая. Как видно из рис. 2, значения плотности распределения весов вершин графов хорошо аппроксимируются прямыми – плотностями степенных распределений с найденными в разделе 3 параметрами. Поэтому можно сделать вывод, что полученные распределения эмпирических данных подчиняются степенным законам с указанными параметрами α.

Также построим функции распределения для рассматриваемых распределений весов вершин графов G_8, G_9, G_{11}, G_{12} и найденных теоретических степенных распределений (рис. 3). Как видно из графиков, эмпирические функции распределения также хорошо приближаются найденными теоретическими. Именно максимумы разниц между этими эмпирическими функциями распределения вершин и найденными

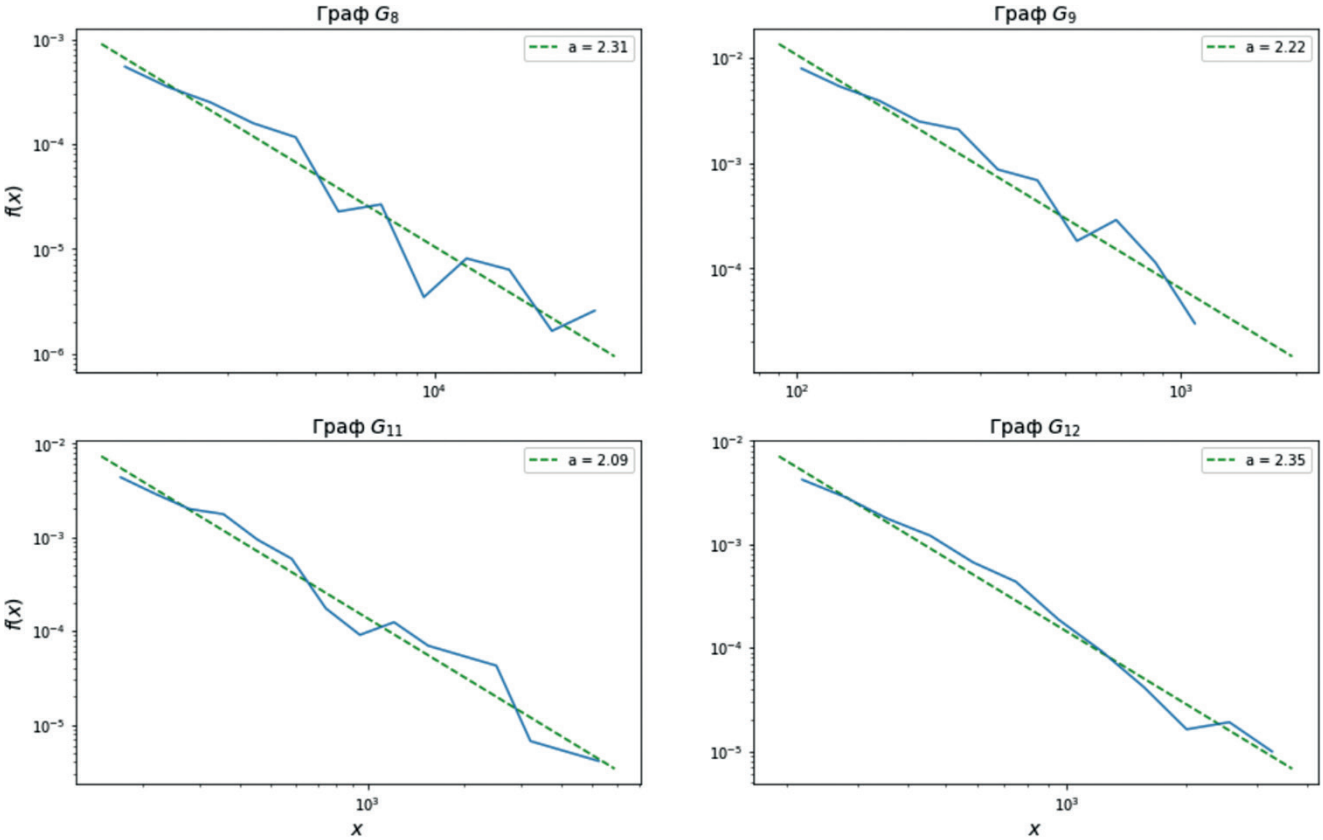


Рис. 2. Синие линии – графики плотности вероятности распределения весов вершин для графов G_8, G_9, G_{11}, G_{12} при весовой функции $F_{1,1}$; Зеленые пунктирные линии – графики плотности вероятности степенных распределений с указанными параметрами α.

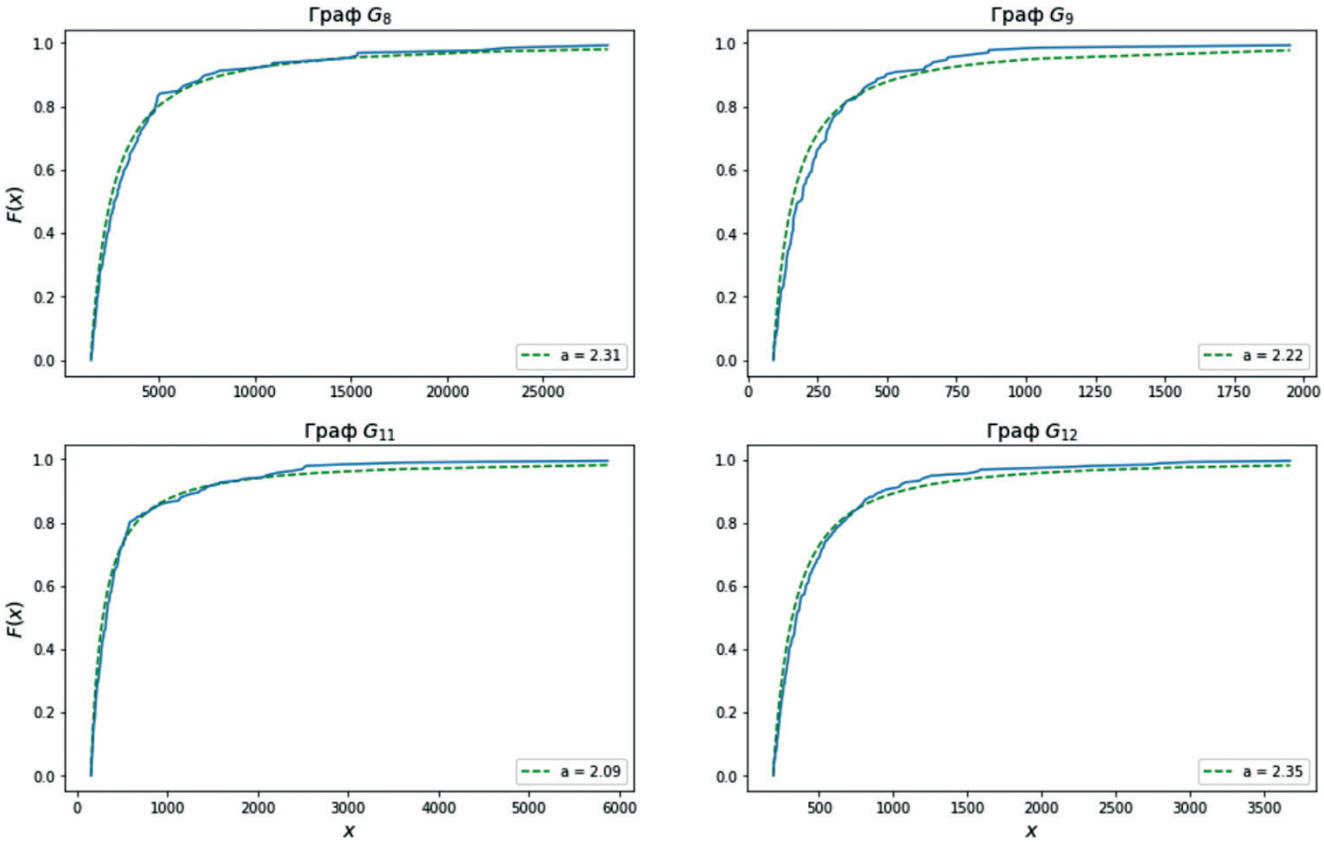


Рис. 3. Синие линии – графики функций распределений весов вершин на графах $G(V,\tilde{E})$ для G_8, G_9, G_{11}, G_{12} при весовой функции $F_{1,1}$; Зеленые пунктирные линии – графики функций распределений степенных распределений с указанными параметрами α.

степенными функциями являются значениями статистик Колмогорова (17), посчитанных в 3 разделе.

Таким образом, наглядно видно, что эмпирические распределения весов вершин для графов G_8, G_9, G_{11}, G_{12} и функции $F_{1,1}$ действительно являются степенными. А сравнение эмпирических и теоретических функций распределения и функций плотности вероятности демонстрирует подтверждение гипотезы о принадлежности выборок весов вершин степенным распределениям с зафиксированными параметрами α.

5. Выводы

В данной статье приведено исследование, посвященное вопросу выбора модели для построения взвешенных графов взаимодействующих объектов сети публичных Telegram-каналов. Были исследованы 12 весовых функций, используемых для построения таких графов. Тестирование проведено для импортированных 15 подсетей, на основе которых были построены 180 графов.

Результаты показывают, что графы, построенные с помощью весовой функции, которая соответствует (U,M,R)-модели в основном имеют распределение весов вершин, близкое к степенному закону. Более того, параметр степенного распределения, наиболее близкого к эмпирическим данным, находится для случая $U = 1$, $M = 2$, $R = 3$ как правило на отрезке [2; 3]. Таким образом, для построенных так графов выполняется свойство безмасштабных сетей.

У графов, построенных при использовании других рассмотренных весовых функций, указанные свойства выполняются реже, что свидетельствует о целесообразности использования (U,M,R)-модели с параметрами $U = 1$, $M = 2$, $R = 3$.

Литература

1. *Fortunato, S., Newman, M. E. J. 20 years of network community detection // Nat. Phys. 2022. № 18. P. 848–850.*
2. *Чеповский А. А. О неявных сообществах на графе взаимодействующих объектов // Успехи кибернетики. 2023. Т.4. № 1. С. 56–64.*
3. *Blöcker, C., Nieves, J. C. & Rosvall, M. Map equation centrality: community-aware centrality based on the map equation. // Appl Netw Sci. 2022. № 7:56. — 24 p. DOI: 10.1007/s41109-022-00477-9*
4. *Rajeh, S., Savonnet, M., Leclercq, E. et al. Comparative evaluation of community-aware centrality measures // Qual Quant. 2023. 57. P. 1273–1302. DOI: 10.1007/s11135-022-01416-7.*
5. *Попов В. А., Чеповский А. А. Модели импорта данных из мессенджера Telegram // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2022. Т.20. №2. С. 60–71.*
6. *Попов В. А., Чеповский А. А. Выделение неявных пересекающихся сообществ на графе взаимодействия Telegram-каналов с помощью «метода Галактик» // Труды ИСА РАН. 2022. Т.72. №4. С. 39–50.*
7. *Чеповский А.А. Анализ графов взаимодействующих объектов. – М.: Национальный открытый университет «ИНТУИТ». 2022. – 270 с.*
8. *Щербакова, Н. Г. Модели сетей с предпочтительным присоединением // Проблемы информатики. 2019. № 3(44). С. 46–61.*
9. *Бадрызлов, В. А., Юдина М. Н. Исследование процессов распространения информации в социальной сети методом имитационного моделирования / Десятая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности «Имитационное моделирование // Теория и практика» (ИММОД-2021): Труды конференции, Санкт-Петербург, 20–22 октября 2021 года / Редакторы Плотников А. М., Долматов М. А., Смирнова Е. П. – Санкт-Петербург: АО «Центр технологии судостроения и судоремонта». 2021. С. 89–94.*
10. *Tereza Nečasová, Ninon Burgos, David Svoboda, Chapter 25 – Validation and evaluationmetrics for medical and biomedical image synthesis / Biomedical Image Synthesis and Simulation // Academic Press. 2022. P. 573–600. DOI: 10.1016/B978-0-12-824349-7.00032-3.*

МОДЕЛЬ СИСТЕМАТИЗАЦИИ КЛАССИФИКАТОРОВ ДЕСТРУКТИВНЫХ И КОНСТРУКТИВНЫХ СОБЫТИЙ ЦИФРОВОГО ПРОСТРАНСТВА

Рыженко А. А.[1], Селезнёв В. М.[2]

DOI: 10.21681/2311-3456-2024-3-113-119

Целью работы *является разработка обобщенной формальной модели систематизации основных классификаторов деструктивных и конструктивных событий инфраструктуры цифрового пространства суверенного государства для организации автономности интеллектуального агента в форме фасета данных.*

Метод исследования: *использование синтаксического представления данных теории информации на стыке модели управления сложными системами и модели информационной безопасности для формализации в виде концептуальной модели.*

Результат исследования: *разработана обобщенная модель систематизации классификаторов деструктивных и конструктивных событий противоборствующей системы в пределах цифрового пространства суверенного государства, позволяющей не только использовать собственные ресурсы для прогнозирования атак и устранения деструктивных элементов на программном уровне, но и привлекать цифровой образ социальной среды как одного из основных элементов для решения задач. В качестве связующего звена в работе предложено использовать интеллектуальных агентов бот-сети, функционал которых предполагает не только теневое взаимодействие с пользовательскими рабочими местами, но и работой с социальной средой непосредственно. Полученная постановка решает актуальную проблему формализации данных – моделирование процессов противодействия внешним деструктивным атакам с распределением функциональных задач, что позволит пересмотреть концепцию собственной безопасности, увеличить стойкость цифровой среды к вероятным негативным воздействиям.*

Научная новизна *заключается в разработке нового элемента концептуального моделирования деструкторов в виде автономных моделей – фасетно-атрибутивного процесса, позволяющего не только адаптивно изменять правила перехода состояний, но и модифицировать собственные параметрические показатели.*

Ключевые слова: *деструктор, моделирование, интеллектуальный агент, фасет, иерархия, правила перехода, автоном, цифровое пространство, система.*

MODEL OF SYSTEMATIZATION CLASSIFIERS OF DESTRUCTIVE AND CONSTRUCTIVE EVENTS IN THE DIGITAL SPACE

Ryzhenko A. A.[3], Seleznev V. M.[4]

The aim of the work *is to develop a generalized formal model for systematizing the main classifiers of destructive and constructive events in the infrastructure of the digital space of a sovereign state to organize the autonomy of an intelligent agent in the form of a data facet.*

Research method: *using a syntactic representation of information theory data at the intersection of a model for managing complex systems and an information security model for formalization in the form of a conceptual model.*

Research result: *a generalized model for systematizing classifiers of destructive and constructive events of an opposing system within the digital space of a sovereign state has been developed, which allows not only to use its own resources to predict attacks and eliminate destructive elements at the program level, but also*

1 Рыженко Алексей Алексеевич, кандидат технических наук, доцент, доцент кафедры информационной безопасности, Финансовый университет при Правительстве РФ, г. Москва, Россия. E-mail: AARyzhenko@fa.ru

2 Селезнёв Владимир Михайлович, кандидат технических наук, заведующий кафедрой информационной безопасности, Финансовый университет при Правительстве РФ, г. Москва, Россия. E-mail: VMSeleznyov@fa.ru

3 Aleksey A. Ryzhenko, Ph.D., Associate Professor, Financial University under the Government of the Russian Federation, Moscow. E-mail: AARyzhenko@fa.ru

4 Vladimir Seleznev, Ph.D., Head of department of information security, Financial University under the Government of the Russian Federation, Moscow. E-mail: VMSeleznyov@fa.ru

to involve the digital image of the social environment as one of the main elements to solve problems. As a connecting link in the work, it is proposed to use intelligent botnet agents, the functionality of which involves not only shadow interaction with user workstations, but also work with the social environment directly. The resulting formulation solves the pressing problem of data formalization - modeling the processes of countering external destructive attacks with the distribution of functional tasks, which will allow us to reconsider the concept of our own security and increase the resistance of the digital environment to possible negative impacts.

***The scientific novelty** lies in the development of a new element of conceptual modeling of destructors in the form of autonomous models – a facet-attributive process, which allows not only to adaptively change the rules of state transition, but also to modify one's own parametric indicators.*

***Keywords:** destructor, modeling, intelligent agent, facet, hierarchy, transition rules, autonomy, digital space, system.*

Введение

Анализ многолетней статистики атак на цифровое пространство организаций и предприятий разного уровня и профиля выявил достаточно формализуемые закономерности, связанные с разными профессиональными категориями социальной среды. Здесь сразу необходимо сделать акцент, что описательных публикаций, содержащих разные аспекты и классификаторы объектов и процессов цифрового пространства в пределах одного государства достаточно много. В частности, можно выделить классификаторы, сопоставимые с направлениями деятельности или с группами документооборота и делопроизводства организаций. Дальнейший синтез и агрегация привели к более укрупненным классам по ключевым позициям: территориальная принадлежность, зональность по видам финансово-денежных отношений, границы в социальной среде и т.д. Каждый укрупненный класс в настоящее время уже имеет достаточно теоретических и практических наработок, позволяющих систематизировать потоковые данные всемирной цифровой среды, ограничивать доступ к информационным ресурсам, публикуемым в разрез с позицией государства (что также отражено в ряде нормативной документации международного уровня, например, «Право на забвение»). Дальнейшее объединение укрупненных классов не привела бы к положительным результатам, в результате, для данных выделенных классов с одной стороны начали образовываться научные школы (что также отражено во многих публикациях), с другой – на практике многие теоретические подходы нашли свое применение. Например, по линии ФСТЭК России такие достаточно сложные документы как модели угроз в организациях вобрали в себя многолетний опыт типов защит от атак. Но также есть и третья сторона, усложняющая процесс внедрения современных формальных моделей на практике – несостыковка позиций основных критериев в обычной среде и в цифровой. Например, территориальную принадлежность государства можно отобразить на географических и политических картах, а границы суверенного государства в цифровом пространстве определить с заданной точностью практически невозможно. Также стоит учесть, что за последние десятилетия основной акцент актуализации методов защиты информации все больше склоняется к границам в социальной среде, что непосредственно связано с массовым развитием методов негативного воздействия социальной инженерии[5]. Как следствие, дальнейшая формализация метода систематизации классификаторов деструктивных и конструктивных событий будет рассматриваться только как новая форма защиты от современных методов воздействия на социальную среду. Здесь необходимо сделать также и второй небольшой акцент – не совсем понятная тенденция отказа во многих публикациях изучать хронологию становления теоретических основ систем безопасности, основанных на практических результатах более чем десятилетней давности. Уже можно отметить ряд последствий, попадающих под «все новое – это хорошо забытое старое» и «лучшее – враг хорошего». В данной публикации в краткой описательной форме представлен результат анализа, как пример необходимости изучать истоки на примерах положительных достижений.

За последние уже 14 лет электронные издания все чаще публикуют результаты независимых классификаций, а также способов и подходов применения исторически устоявшихся психологических методологий воздействия на массовой сознание социума. Развитие современных информационных систем и технологий способствовало формированию принципиально новых методов воздействия, сочетавшие модели массового влияния, но с учетом персонификации каждого объекта воздействия в индивидуальном порядке. Устоявшиеся за последние 24 года

5 Социальная инженерия: анализ и методы противодействия. – режим доступа: https://cisoclub.ru/socialnaja-inzhenerija-analiz-i-metody-protivodejstvija/

методы последовательного воздействия на группы субъектов с целью достижения конкретной цели стало одним из основных направлений социальной инженерии. Например, телефонное мошенничество, «основанное» и описанное Кевином Митником в известном издании «Искусство обмана» [1]. Так широко разрекламированный в 90-е годы данный тип мошенничества развился в корпоративное мошенничество, появились новые формы воздействия на организации с использованием четко сформированных сценариев, алгоритмов и прочих последовательных этапов выполнения целевых задач. Массовые атаки на информационные ресурсы организаций, как правило, готовятся месяцы. Основные объекты предварительных (подготовительных) атак – простые сотрудники организаций, имеющие доступ к базам данных и прочей документации систем документооборота. С одной стороны, данная тенденция роста деструктивных воздействий должна волновать исключительно атакуемые объекты, с другой – явная тенденция последних лет массовой переориентации атак на критически важные объекты государственного уровня уже преподносит негативные последствия и недвусмысленно намекает на активное внедрение новых форм противоборства.

В результате многолетнего всестороннего анализа исторических фактов получено множество зависимостей прямых сценариев «*причина – фактор – сценарии*» и обратных «*цель – задачи – сценарии*». Появилась возможность формирования семантических связей, позволяющих автоматизировать процесс перехвата злоумышленников на разных этапах воздействия на социальную среду уровня цифрового пространства. Например, использование на практике разных методов доступа к конфиденциальной информации позволило сделать базу правил действий (в том числе и для разрабатываемого интеллектуального агента фишинговой системы) [2]. Применяя доступные каналы доступа к информации (социальные сети, мессенджеры, чаты и т.д.) агент собирает информацию о сотрудниках, создает информационные суррогаты биоинформации, готовит сценарии атак на информацию организаций с использование аппаратных и программных ресурсов сотрудников данных организаций (пассивные агенты).

Особенностью данной работы является разработанная модель обратной задачи элемента бот-сети, а также алгоритм адаптивного внедрения в действующую информационную систему без ущерба основному жизненному циклу как самой системы, так и окружающей цифровой среды (пространства) – интеллектуального агента в форме самостоятельно модифицируемого деструктора в условиях частичной анонимности.

1. Предыстория и связанная с этим работа

У данной модели, как и у множества аналогичных, достаточно продолжительная история с неоднозначными путями развития. Одним из ключевых исторических моментов стал учет юридического опыта многих государств с целью выявления базовых сценариев противоборства компьютерным атакам на социальную среду. Например, в работе[6] автор рассматривает ряд вопросов, заложивших пути дальнейшего развития такого важного направления, как самозащита в организованных сетях, где социум мог (на тот момент времени) дистанционно общаться по разным вопросам на расстоянии без очного участия. В продолжении данного материала хочется отметить работу[7], где автор приводит альтернативные точки зрения на методологии судебных компьютерных исследований. Данный фактор вызван существенными изменениями даже на тот момент в сфере компьютерных преступлений с использованием сетей общего доступа. В России данный период ознаменовался вступлением в силу первых достаточно серьезных статей уголовного кодекса, предполагающих наказание за совершенные преступления в цифровой среде.

Необходимо отметить, что преступный мир не стоял на месте и развивался в 2000-х годах намного быстрее, чем могли себе это представить правоохранительные органы того времени. Например, достаточно известное эссе прародителя всемирной сети Интернет Билла Джоя в 2000 году заложило принципиального новое направление в сфере деструкторов – *метаморфы* и *полиморфы*. Особенностью данных конструкций была возможность саморазвития за счет использования трех практически независимых составных частей: голова, тело и хвост. Достаточно серьезные исследования возможного противоборства данной разновидности деструкторов приводили к одной общей мысли: *уничтожить данный вид деструктора практически невозможно, а проводить аудит действий и прогнозировать возможные негативные процессы вполне возможно*.

Тем не менее, благодаря первому полученному уже многолетнему опыту менялись сценарии противоборства, модифицировались алгоритмы и методики, о чем было отмечено во множестве публикаций. На тот момент многие разработчики уже задумались о возможности использовать дополнительные встроенные модули в систему обозревателей Интернет, позволяющие использовать антифишинговые

6 Bénichou, D., Lefranc, S. Introduction to Network Self-defense: technical and judicial issues. J Comput Virol 1, 24-31 (2005). https://doi.org/10.1007/s11416-005-0006-5

7 Broucek, V., Turner, P. Winning the Battles, Losing the War? Rethinking Methodology for Forensic Computing Research. J Comput Virol 2, 3-12 (2006). https://doi.org/10.1007/s11416-006-0018-9

панели с использованием функционала *cookies*, что применяется и по настоящее время[8].

Следующий год был ознаменован тем, что данный тип деструкторов перешел на новый уровень, появилась новая функция, предполагающая размножение вирусов в полуавтономном режиме. Также деструкторы научились скрывать свои тела в других программах как часть кода, и в других процессах доверенной зоны, а две другие составляющие (голова и хвост) стали независимы друг от друга, но вполне объединяющиеся при обнаруживании в пределах одной системы. Описанные процессы и множество других негативных факторов отмечены в ряде работ[9].

В 2010 году разработчики методов и моделей противоборства вирусным атакам обратили внимание на интеллектуальные системы, способные, как и вирусы *полиморфики*, самостоятельно развиваться в полуавтономном режиме. Многие публикации того времени отразили приоритетные направления исследований, а также предполагаемые результаты. Необходимо сразу отметить, что прогнозы десятилетней давности сбылись и уже применяются на практике. Например, авторами представлен новый метод интеллектуального анализа данных и пример использования нейронных сетей для обнаружения определенного типа деструкторов[10]. Также не забыты и классические методы, показавшие свою эффективность и в других направлениях моделирования. При этом исследования процессов новых модификаций деструкторов продолжились, получая все новые результаты.

В 2013 году исследователи обратили внимание в существенное изменений вирусов метаморфов. Новые энтропийные функции позволяли вносить некоторую избыточность в тело, что позволяло активировать новый функционал встроенного самоконтроля. Благодаря новой технологии метаморфы могли прятаться и становиться псевдонимны. Данный процесс стали позже использовать при хранении персональных данных. Эксперты также провели достаточно интересное исследование, позволяющее дать однозначную оценку в возможных путях развития вредоносных деструкторов, а также предполагаемые механизмы обнаружения. Естественным развитием стало внедрение модифицированной технологии обнаружения новой формы вирусных атак. Исследование результатов деятельности внедренных интеллектуальных агентов позволили выявить структуры новой формы метаморфа, что позволило в последующем разработать новые инструменты противоборства[11].

На данном этапе развития противоборства в цифровой среде можно выделить новую особенность, связанную с перенаправлением части исследовательских ресурсов в сторону защиты от новых методов социальной инженерии, напрямую связанных с деятельности вредоносного полиморфного вируса. Многими исследователями отмечено, что классическая форма прямой атаки на социум, с использованием сети Интернет стало частью прошлого. Современные полуавтономные сети модифицирующихся вирусов, собирающие данные о пользователях как части общей системы – новый опасный фактор. Как следствие, текущий анализ целевых атак на организации должен быть более комплексным с учетом возможных вариаций и альтернативных вариантов решений. Новые исследования в данном направлении привнесли в существующие алгоритмы достаточно эффективные методы выбора альтернатив сценариев противоборства метаморфам с учетом опыта многолетних ошибок. Например, ошибок в существующем программном обеспечении[12].

Развитие интеллектуальных систем не стояло на месте и продолжало развиваться в разных направлениях противоборствующих систем. Например, интересный подход обнаружения скрытых атак в потоке информации, описанный в [3], а также возможность использования результатов исследований, способствовали выделению в последующих исследованиях нескольких типов вредоносного программного обеспечения. Также получило развитие направление выявления скрытых угроз как для полиморфов, так и для метаморфов. В результате, можно отметить совершенствование существующих моделей злоумышленник / защитник для обучения кибербезопасности интеллектуальных моделей, в том числе и моделей противоборства социальной инженерии [4].

В 2020 году разработанный ранее подход противоборства видоизменяющимся вирусным атакам получил системный подход, о чем было отмечено в работе [5]. Разрабатываемые на основе новых моделей семантические алгоритмы нейронных сетей, способных выявлять формирование бот-сетей на разных этапах положили новое направление исследований, где роль интеллектуальных агентов стала иметь важное значение [6]. Также хочется отметить, что развитие моделей раннего обнаружения негативного фактора, приводящего к деструктивным

8 Li, L., Helenius, M. Usability evaluation of anti-phishing toolbars. J Comput Virol 3, 163–184 (2007). https://doi.org/10.1007/s11416-007-0050-4

9 Jacob, G., Filiol, E. & Debar, H. Functional polymorphic engines: formalisation, implementation and use cases. J Comput Virol 5, 247-261 (2009). https://doi.org/10.1007/s11416-008-0095-z

10 El-Bakry, H.M. Fast virus detection by using high speed time delay neural networks. J Comput Virol 6, 115-122 (2010). https://doi.org/10.1007/s11416-009-0120-x

11 Mezzour, G., Carley, L.R. & Carley, K.M. Longitudinal analysis of a large corpus of cyber threat descriptions. J Comput Virol Hack Tech 12, 11-22 (2016). https://doi.org/10.1007/s11416-014-0217-8

12 Tripathi, N., Hubballi, N. Detecting stealth DHCP starvation attack using machine learning approach. J Comput Virol Hack Tech 14, 233-244 (2018). https://doi.org/10.1007/s11416-017-0310-x

последствиях, стало широко использоваться и в социальных сетях [7].

Развитие не стоит на месте и продолжает совершенствоваться. За последующие 2 года получены достаточно интересные результаты в плане исследований внутренних процессов метаморфов с целью раннего прогнозирования возможных деструктивных действий на социальную среду [8].

С другой стороны, многолетний анализ приводит к неоднозначному выводу несуществования единой модели противоборства как самим проявлениям полиморфа и метаморфа, так и адаптивной бот-сети, состоящей из нестабильных интеллектуальных агентов. Также данному обстоятельству способствует быстро развивающееся направление анонимизации агентов, хранящих в себе саморазвивающиеся процедурные алгоритмы. Как следствие, в данной работе предлагается рассмотреть процессную модель формирования правил выбора альтернативных решений с множественным решением. Данная технология достаточно хорошо показала себя на практике при разрешении коллизий разного уровня в достаточно плотном пространстве, т.е. при множественном анализе системы и как целое, и как множество несвязанных автономов одновременно.

2. Новый подход формализации данных, основанный на алгебре процессов

В качестве схемы организации данных первичной обработки для построения сценариев, как было представлено ранее, например [9], используем:

- *базу правил развития событий* (обратное дерево решений) на основе сформированных алгоритмов многолетнего опыта исследователей для определения сценариев атак с использованием разветвленных альтернативных решений, представленных по ссылкам на публикации в предыдущей части данной статьи. Например, публикации по выявлению атак в социальных сетях позволили сформировать базовые правила подготовительного и начального этапов;
- *базы ассоциаций*, позволяющие порождать интеллектуальных агентов (прямое дерево решений) под конкретные задачи и хранящие целевое предназначение каждого агента в пассивном, теневом режиме вплоть до активной фазы. Например, при краже личности и формировании цифрового образа человека первым этапом является анализ социальной сети и сетевой активности в известных мессенджерах. Порождаются (формируется однозадачный код) агенты выполняющие четко определенные задачи поиска в определенное время (по таймеру или будильнику). Как правило такие агенты закладываются на независимые ресурсы всемирной сети в анонимной форме.

Вариант построения деревьев рассмотрен при моделировании системы организации данных бот-сети [10]. Далее процесс моделирования разбит на две последовательные части:

1) моделирование процессов перехода состояний по ветвям деревьев в условиях множественного выбора и множества решений;
2) моделирование системы организации данных адаптивных моделей бот-сетей в условиях начальной неопределенности.

3. Моделирование системы организации процессов многокритериальной модели дискретного пространства данных

Как уже было упомянуто ранее, анализ исходных формальных моделей, приведенных в первой части статьи позволил сформировать связи между конструктивной частью системы и деструктивной в виде продукционных правил в алгебраической форме. Особенности построения правил в системе процессов алгебры мультимножеств расшифрованы в публикациях, где приведен анализ процесса обработки данных систем документооборота для организаций финансового сектора и промышленной среды, а также формам формализации систем поддержки управления [11]. Дальнейшее моделирование потребовало увязать несколько независимых форм теории управления системами в единый формат [12]. Укрупненное схематичное представление полученного результата представлено на рисунке 1.

Принципиальное отличие данного представления обобщенной модели процессной обработки данных состоит в следующем:

- выявленные деструкторы по произвольной существующей классификации не представлены в форме возможных последствий, а также возможных форм защиты от негативных последствий. Комбинация деструктора и атакуемого информационного ресурса представляет пару исходных данных для узловой точки дерева событий. Цель – выявление возможных путей развития, построение сценария с альтернативными вариантами развития. Данная методология позволяет не анализировать деструктор не как отдельный элемент, а как часть самоорганизованной системы бот-сети;
- параллельно ведется анализ множества действующих деструкторов, а также конструкторов адаптивной защиты. Текущее состояние прописывается в фасет текущего состояние. Как было отмечено в предыдущих изданиях, особенностью фасета является граничность каждой ячейки. На текущий момент используются ячейки с шестью границами, в отличии от классической ячейки с четырьмя границами. За счет внесения искусственной избыточности удалось избежать коллизии при неопределенности решения;

– дискретное пространство баз ассоциаций позволяет упростить процесс обработки за счет упрощенной формы представления данных по аналогии трансформации битового представления при переходе от хранения в 8-битном формате в 6-битный для транспортировки по каналам связи. За счет того, что продукционные правила ограничены в символах, сформирован искусственный алфавит.

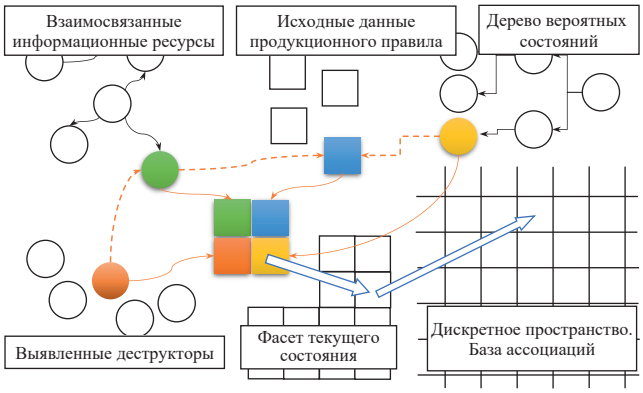


Рис. 1. Схема взаимодействия полуавтономной противоборствующей бот-сети интеллектуальных агентов и деструкторов цифровой инфраструктуры

Для формирования связей между двумя классами баз ассоциаций деструкторов и конструкторов используются оси идентификаторов матричного представления (рис. 2). Для визуального представления можно сформировать аффинную систему координат, где две плоскости – базы ассоциаций.

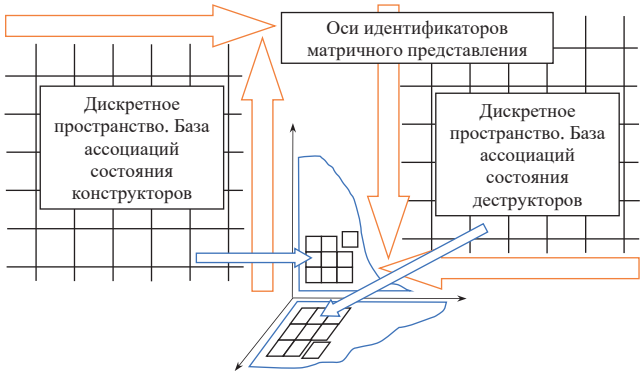


Рис. 2. Использование независимых фасетов организации данных баз ассоциаций интеллектуальных агентов бот-сети

Полученная кубическая матрица для связи двух граней состояний в форме дискретного пространства невзаимодействующих баз ассоциаций состояний конструктора / деструктора системы оснащается третьей составляющей – иерархия преобразования альтернативных состояний (рис. 3) [13].

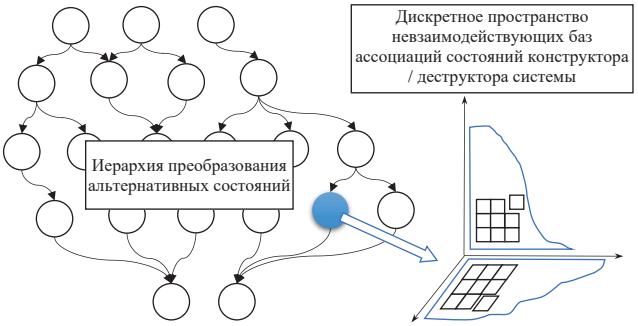


Рис. 3. Использование деревьев для формирования связей между состояниями интеллектуальных агентов бот-сети

Алгоритм формирования продукционного процессного правила бот-сети следующий:

– фасет деструкторов добавляет к каждому внедренному правилу координаты ячейки, определяет вероятных соседей по схожим признакам (первая часть решения);
– фасет конструкторов добавляет к каждому определенному правилу координаты ячейки, определяет схожие с деструктором по текущему состоянию признаки (вторая часть решения);
– производится поиск правила перехода к вероятному следующему состоянию в базе правил, строится узел дерева преобразования состояний (третья часть решения) (рис. 3).

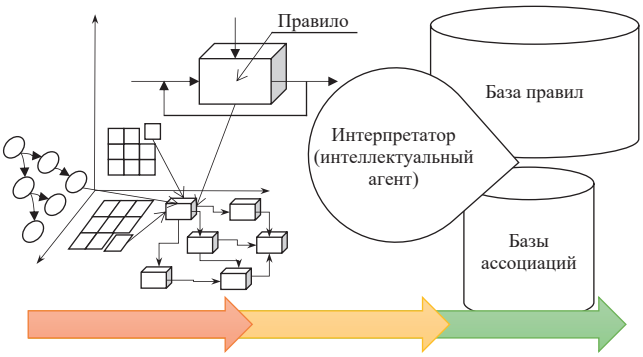


Рис. 4. Обобщенная схема интеграции противоборствующей бот-сети в единую цифровую среду

Как и в классическом представлении целевых иерархических деревьев, каждое алгебраическое правило снабжается индикатором состояния по принципу светофор. Данная методология используется в связи со своей простотой применения в рисковых моделях. Как следствие, каждый узел дерева снабжается рисковым двухкритериальным коэффициентом опасности в процентном соотношении и коэффициентом последствий в аналоге денежного эквивалента.

Заключение

Представленная методология использования разработанной модели широко описана во многих публикациях [14, 15], прошла апробацию в разных сферах профессиональной деятельности при формализации и дальнейшем применений результатов моделирования, а также в более 10 диссертационных исследований не только в пределах РФ. Успешное применение при составлении продукционных правил в упрощенной алгебраической форме теории мультимножеств на множестве процессов еще раз подтверждает тот факт, что отечественные разработки, совершенствующие семантическую составляющую теории информации, позволит перейти к более качественному уровню не только для защиты информации на уровне суверенного государства, но и для предупреждения возможных атак со стороны предполагаемых и уже действующих оппонентов[13].

13 Статья подготовлена по результатам исследований, выполненных за счет бюджетных средств по государственному заданию Финуниверситета

Литература

1. *Кевин Митник «Искусство обмана». – режим доступа: https://remarx.ru/media/books/ iskusstvo_obmana_mitnikpdf.pdf*
2. *Рыженко А. А. Умная бот-сеть или модель интеллектуального деструктора // Вопросы кибербезопасности. 2023. № 5(57). С. 60–68. DOI: 10.21681/2311-3456-2023-5-60-68*
3. *Gibert, D., Mateu, C., Planes, J. et al. Using convolutional neural networks for classification of malware represented as images. J Comput Virol Hack Tech 15, 15-28 (2019). https://doi.org/10.1007/s11416-018-0323-0*
4. *Bernardeschi, C., Domenici, A. & Palmieri, M. Formalization and co-simulation of attacks on cyber-physical systems. J Comput Virol Hack Tech 16, 63-77 (2020). https://doi.org/10.1007/s11416-019-00344-9*
5. *Jain, M., Andreopoulos, W. & Stamp, M. Convolutional neural networks and extreme learning machines for malware classification. J Comput Virol Hack Tech 16, 229-244 (2020). https://doi.org/10.1007/s11416-020-00354-y*
6. *Rahman, R.U., Tomar, D.S. Threats of price scraping on e-commerce websites: attack model and its detection using neural network. J Comput Virol Hack Tech 17, 75-89 (2021). https://doi.org/10.1007/s11416-020-00368-6*
7. *Reddy, V., Kolli, N. & Balakrishnan, N. Malware detection and classification using community detection and social network analysis. J Comput Virol Hack Tech 17, 333-346 (2021). https://doi.org/10.1007/s11416-021-00387-x*
8. *Ebrahim, M., Golpayegani, S. A. H. Anomaly detection in business processes logs using social network analysis. J Comput Virol Hack Tech 18, 127-139 (2022). https://doi.org/10.1007/s11416-021-00398-8*
9. *Рыженко А. А., Рыженко Н. Ю. Интеллектуальные деструкторы и мобильные банковские клиенты / Актуальные проблемы и перспективы развития экономики: Труды XXI Международной научно-практической конференции. Симферополь-Гурзуф, 20–22 октября 2022 год. / Под ред. д.э.н., д.пед.н., профессора Н. В. Апатовой. – Симферополь: Издательский дом КФУ им. В. И. Вернадского, 2022. – с. 241-242.*
10. *Рыженко А. А. Модифицированный алгоритм вируса полиморфика как основа деструктора информационной среды / Информатика: проблемы, методология, технологии: сборник материалов XVIII международной научно-методической конференции: в 7 т. / под редакцией Н. А. Тюкачева; Воронеж, Воронежский государственный университет, 14-15 февраля 2019 г. – Воронеж: Издательство «Научно-исследовательские публикации» (ООО «Вэлборн»), 2019. – Т. 5. – С. 857–861.*
11. *Рыженко А. А. Модель вложенной пирамиды системы управления безопасностью информационного пространства госкорпорации / Противодействие терроризму и экстремизму в информационных системах: сборник научных статей Всероссийской конференции – М.: Московский университет МВД России имени В. Я. Кикотя, 2020. – с. 65–69.*
12. *Рыженко А. А., Рыженко Н. Ю. Безопасность информации цифровой экономики / Актуальные проблемы и перспективы развития экономики. Труды Юбилейной XX Всероссийской с международным участием научно-практической конференции. Симферополь, 2021. С. 289–291.*
13. *Рыженко А. А. Фасетно-иерархическая модель как альтернатива существующим моделям систем поддержки управления / Управление информационными ресурсами. Материалы XIX Международной научно-практической конференции. Минск, 2023. С. 37-38*
14. *Любавский А. Ю. О необходимости развития алгоритмического мышления следователей в контексте расследования киберпреступлений / Проблемы противодействия киберпреступности. Материалы международной научно-практической конференции. Москва, 2023. – с. 105–108.*
15. *Любавский А. Ю. Актуальные вопросы обеспечения безопасности персональных данных в сети интернет / Обеспечение информационной безопасности: вопросы теории и практики. Сборник статей Всероссийской научно-практической конференции. Науч. редакторы Г. Г. Камалова, В. Г. Ившин, Г. А. Решетникова. Ижевск, 2023. – с. 39–45.*

МЕТОДОЛОГИЯ ИДЕНТИФИКАЦИИ АВТОРА ТЕКСТА ДЛЯ РЕШЕНИЯ ЗАДАЧ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Романов А. С.[1]

DOI: 10.21681/2311-3456-2024-3-120-128

Цель работы: *создание методологии идентификации автора текстовой информации, включая естественно-языковые тексты и исходные коды программ, для решения задач информационной безопасности.*

Объектом исследования *является печатный текст и его характеристики.*

Предметом исследования *являются характеристики текста, описывающие авторский стиль, методы и алгоритмы машинного обучения, предназначенные для работы с естественно- и искусственно-языковыми текстами.*

Методы исследования *включают методы теории множеств, математической статистики, вычислительного эксперимента и методы искусственного интеллекта.*

Научная новизна: *предложена комплексная методология идентификации автора текста, учитывающая особенности естественно- и искусственно-языковых текстов, а также предложена модель создания текста автором в киберсреде, впервые учитывающая семантические особенности и информативные признаки текста на разных уровнях иерархического анализа, специфику среды, атрибуты автора и вид деятельности по созданию текста.*

По результатам исследования *предложена методология идентификации автора естественно-языкового текста и исходных текстов программ для решения задач информационной безопасности в виде комплекса методов, моделей и алгоритмов, агрегирующий имеющийся опыт. Методология является универсальной для решения задач информационной безопасности, связанных с классификацией текстов.*

Ключевые слова: *интеллектуальный анализ текста, семантика, машинное обучение, исходный код, атрибуция.*

METHODOLOGY FOR IDENTIFYING THE AUTHOR OF TEXT INFORMATION FOR SOLVING CYBERSECURITY TASKS

Romanov A. S.[2]

The goal of article: *the creation of a methodology for identifying the author of textual information, including natural language texts and program source codes, is aimed at solving information security issues.*

The object of study: *printed text and its characteristics.*

The subject of study: *characteristics of text that describe the author's style, methods, and machine learning algorithms designed for processing both natural and artificially-generated texts.*

The research methods: *set theory methods, mathematical statistics, computational experiments, and methods of artificial intelligence*

Scientific novelty: *for the first time, a comprehensive methodology for identification of a text's author and a model for text creation by an author in a cyber environment have been proposed. The proposed methodology considers features of both natural and artificially-generated texts. An introduced model takes into account semantic features and informative characteristics of the text at different levels of hierarchical analysis, specifics of the environment, author attributes, and the nature of activities involved in creating the text.*

1 Романов Александр Сергеевич, кандидат технических наук, доцент, доцент кафедры Комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС), Томский государственный университет систем управления и радиоэлектроники, Томск, Россия. ORCID: 0000-0002-2587-2222. Scopus Author ID: 57221288963. E-mail: alexx.romanov@gmail.com

2 Romanov Aleksandr Sergeevich, Candidate of Technical Sciences, Associate Professor, Department of Complex Information Security of Electronic Computing Systems, Tomsk State University of Control Systems and Radioelectronics, Tomsk, Russia. ORCID: 0000-0002-2587-2222. Scopus Author ID: 57221288963. E-mail: alexx.romanov@gmail.com

Results obtained: *a methodology has been proposed for identifying the author of a natural language text and program source codes to address information security challenges. The methodology includes a set of methods, models and algorithms that aggregate existing research experience. The methodology is universal for solving information security issues related to text classification.*

Keywords: *text mining, semantics, machine learning, source code, attribution.*

Введение

Один из видов нарушений в киберпространстве – нарушение авторских и смежных прав на текстовые произведения, которое может выражаться, например, в присвоении текста другого человека с целью получения материальной выгоды или попытке выдать авторство созданного текста за авторство другого лица. Методы определение авторства позволяют выявить подобные нарушения и установить личность создателя текста. Атрибуция текстов является важной проблемой в компьютерной лингвистике, журналистике, а также в криминалистике, где знание истинного автора анонимного текста (например, предсмертной записки) может облегчить и ускорить работу правоохранительных органов. Таким образом проблема идентификации автора текста для защиты интеллектуальной собственности является важной задачей информационной безопасности.

В числе атрибутов автора выделяют пол, возраст, образование, профессию, личностные качества и др. Первичными считают гендерный признак и возраст, потому что их конкретизация позволяет установить остальные атрибуты (вторичные) и сузить круг кандидатов при определении автора. Совокупность атрибутов формирует уникальную языковую личность. Одним из направлений практического применения методик определения пола и возраста является криминалистика, где важно определение психологического портрета преступника и профилирование автора. В контексте проблем информационной безопасности подобные методики являются основой для мониторинга социальных сетей для выявления информации, пропагандирующей нетрадиционные сексуальные отношения, педофилию, смену пола. Важной проблемой также является недопущение детей и подростков до запрещенного или шокирующего контента, который имеет возрастное ограничение «18+», либо ограничение общения с целью предотвращения педофилии. Определение гендерной принадлежности автора текста актуально поскольку решением Верховного Суда Российской Федерации от 30.11.2023 по делу № АКПИ23-990С «движение ЛГБТ» признано экстремистской организацией и его деятельность запрещена на территории России. Таким образом определение пола и возраста по тексту для дальнейшего выявления признаков пропаганды ЛГБТ и педофилии является важной задачей информационной безопасности, актуальным является также вопрос создания программного обеспечения для мониторинга социальных сетей.

Методы определения однородности авторского стиля сообщений можно использовать для продленной аутентификации [1] в социальной сетях и мессенджерах, обнаружения аномалий и необычных паттернов в потоке текстовых данных пользователей сети Интернет. Таким образом задача идентификации автора сообщений в сети Интернет и продленная аутентификация пользователя социальных сетей на основе текста является важной задачей информационной безопасности, имеющей существенные особенности в методологическом плане.

Анализ и определение авторства текста с учетом эмоциональной составляющей имеют особую важность в контексте борьбы с экстремизмом и терроризмом [2–5]. В современном информационном обществе социальные сети и онлайн-платформы стали пространством для распространения экстремистских и радикальных идей. Опасность заключается в том, что Интернет-платформы предоставляют экстремистам доступ к широкой аудитории и возможность быстрой и масштабной пропаганды своих идей. Это может воздействовать на уязвимых или подверженных влиянию лиц, включая молодежь, подстрекая их к насилию, террористическим действиям или участию в экстремистских организациях. Мониторинг этой информации в определенные моменты времени и выявление лиц, имеющих целью совершение злонамеренных действий, становится актуальной практической задачей противостояния террористической угрозе и защиты государства. Таким образом анализ настроения автора, определения эмоциональной окраски текста, деструктивных, а также текстов экстремистской направленности, запрещенных законодательством Российской Федерации, является важной задачей информационной безопасности.

Решение задачи определения автора программного кода являются критически значимыми для обеспечения безопасности в цифровой среде. Это связано с тем, что подавляющее большинство технологий, а также программных систем и комплексов, упрощающих профессиональную и повседневную деятельность человека, подвержены

сбоям. Подобные проблемы могут возникать по ряду причин, например, в результате ошибок разработчиков (при проектировании, реализации и/или внедрении), неправильной эксплуатации пользователями, неполадок смежных систем. Однако наибольшую угрозу несут сбои, происходящие ввиду преднамеренного и/или злоумышленного вмешательства. Несмотря на то, что деятельность по созданию, использованию и распространению вредоносного программного обеспечения запрещена на законодательном уровне и закреплена в ст. 272, 273, 274 Уголовного Кодекса Российской Федерации, технические средства, обеспечивающие эффективное и своевременное установления авторства исходного кода в рамках компьютерных экспертиз, на 2024 год отсутствуют. Анализ исходных кодов программ на предмет авторства осуществляется специалистами в области компьютерной криминалистики вручную или с использованием малоэффективных для данной задачи средств текстового анализа. Таким образом, задача определения автора-вирусописателя представляют особую важность для информационной безопасности.

Актуальной задачей информационной безопасности становится проблема создания и усовершенствования методик, учитывающих способы сокрытия авторского стиля (обфускация) и имитации авторского стиля [6], а также генеративных моделей, позволяющих автоматически генерировать тексты на основе глубоких моделей, обученных на больших текстовых корпусах (GPT). В связи с этим возникает необходимость в проведении дополнительных исследований, направленных на оценку устойчивости методов определения авторства текста к такого рода атакам.

Анализ предметной области

В настоящее время наблюдается повышенный интерес к количественным методам анализа текстовой информации на основе слабо контролируемых человеком характеристик текста, общих для всех авторов. С развитием методов текстового анализа, в работах по определению авторства начинает превалировать использование семантической информации о тексте [7], наряду с лексическими, морфологическими и синтаксическими признаками.

Диссертационная работа Москина Н. Д. [8] и связанные с ней исследования [9] фокусируются на разработке и модернизации теоретико-графовых моделей, использующих технологию Graph Neural Network (GNN), для определения авторства текстов. В исследовании анализируются 500 текстов неопределенного авторства и более 800 произведений русских классиков. Основные методы включают агрегацию графов, учет их иерархичности, нечеткости и темпоральности, а также использование метрик, таких как максимальный общий подграф. Проверялись гипотезы о различиях в структурных характеристиках графов разных жанров. Ансамбль моделей был разработан для описания языковой структуры текстов. Для анализа использовались такие методы, как рекуррентная НС, сеть долгой краткосрочной памяти (LSTM), Transformer, дерево решений, SVM, деревья решений (RF). Наилучший результат показала модель Transformer с точностью 97%, в то время как дерево решений показало минимальную эффективность в 43%.

В диссертационной работе Огорелкова И. В. [10] исследуются гендерные различия в русскоязычных политических текстах. Общий корпус текстов состоит из 1000 произведений, разделенных на мужской и женский корпусы по 500 текстов каждый. Исследование включает анализ четырех основных групп признаков: смысловых, текстологических, языковых и психолингвистических, дополненных лексическими и синтаксическими особенностями. В работе выделено 20 ключевых признаков [11], специфичных для мужской и женской письменной речи, такие как использование определенных союзов, частиц, местоимений и вводных слов. Эти признаки затем анализируются для определения пола автора текста. Заключительный этап включает оценку информативности каждого признака и окончательное определение гендерной принадлежности автора. Выводы исследования указывают на характерные различия в стиле мужской и женской речи, такие как лаконичность и аргументированность для мужчин, против многословия и эмоциональности для женщин.

В диссертации Сбоева А. Г. [12] и соответствующей статье [13] обсуждается методика определения пола и возраста автора русскоязычного текста на основе морфологических, синтаксических признаков, n-грамм, токенов, частей речи, эмоциональных признаков и эмбеддингов. Использованы методы глубокого обучения, включая сиамскую нейронную сеть и оригинальную архитектуру SyntGraphLSTM, с моделью представления текста TF-IDF и классификаторами SVM и RF. Корпус состоял из 1850 контролируемых и 41624 реальных текстов из социальных сетей, включая 4332 текста с искаженным полом и 13632 с искаженным возрастом. Методика показала точность в 86% по метрике F1 для определения пола, 64% при намеренных искажениях, 48% для определения возрастной группы (выше случайного угадывания на 15%) и 44% для распознавания искаженного возраста, при этом направление искажения определялось с точностью 80%. Возрастные группы включали 18–23, 24–29 и старше 30 лет.

В диссертации Давыдовой Ю. В. [14] исследуются методы мониторинга контента в социальных сетях с целью реализации превентивных мер пропаганды криминализации. Методы включали технику специализированного текстового поиска на основе динамического программирования и деревьев

решений, новый подход, основанный на семантическом анализе для определения жаргонизмов и неологизмов, тематической лексики пропагандистов криминала, моделирование ошибок на основе гибридной модели, сочетающей лингвистические правила и статистические данные для присвоения веса различным типам текстовых ошибок. Данные для исследования включали федеральные и региональные базы данных правоохранительных органов и платформы социальных сетей. Был разработан прототип программного обеспечения, который апробирован в реальных сценариях для мониторинга и анализа социальных сетей на предмет незаконной деятельности. Предложенный подход продемонстрировал точность 95% при обнаружении противозаконной информации.

В диссертационной работе Андреева И. А. [15] предложена методика построения социального портрета пользователя в рамках подбора кадров с учетом материальных. профессиональных и социальных рисков работодателя. Предложен подход к унификации и агрегации данных из различных социальных сетей, сопоставления профилей пользователей разных социальных сетей, обработка слабоструктурированных данных. Целью работы является формирование психоэмоционального портрета пользователя на основе тонального анализа созданных им текстов и информации из профилей социальных сетей. Набор данных включал более 2,5 миллионов сообщений. Классификация осуществлялась на основе категоризации текстов по 10 эмоциям. При формировании вектора признаков был расширен словарь WordNet-Affect, отдельно проводился анализ эмотиконов. Для проведения экспериментов были отобраны 100 пользователей, имеющих 1 и более аккаунт в социальных сетях. Максимальная точность 87% была достигнута моделью BERT, классические методы машинного обучения (SVM LR, RF) оказались менее эффективны, при их использовании точность составляла не более 65% при использовании SVM.

В диссертации Стремоухова В. Д.[3] рассматривается задача определения авторства бинарного кода, особенно вредоносных программ. Исследование включает применение математического анализа и статистики, таких как Марковские процессы и корреляционный анализ. Стремоухов предлагает модели, в том числе на основе сжатия данных и относительной энтропии, которые анализируют объединение анонимного кода с известными образцами для оценки степени сжатия. Другая модель использует матрицу переходных вероятностей Маркова и способна отсеивать неинформативные части кода. Модели апробированы на исполняемых файлах win32 и вирусных коллекциях «Лаборатории Касперского», показав эффективность с точностью до 100%. Однако, метод требует единообразия в языке программирования, платформе и компиляторе, что ограничивает его применение в сложных случаях.

Перечисленные исследователи успешно решали частные задачи атрибуции, не связанные с обеспечением информационной безопасности напрямую. Однако проблема систематизации существующих подходов к идентификации автора текста, а также разработки комплексной методологии, позволяющей эффективно решать взаимосвязанные задачи авторства в интересах национальной безопасности страны, остается открытой.

Методология идентификации автора текстовой информации

На основе проведенного анализа разработана обобщенная методология идентификации автора текстовой информации для решения задач информационной безопасности (рис. 1).

Представленная на рис. 1 методология оперирует множеством взаимосвязанных моделей, методов и алгоритмов для анализа естественно- и искусственно-языковых текстов и включает:

1. Модель создания автором текста в киберсреде. Ключевая модель методологии, описывающая процесс создания текстов авторами с учетом особенностей и ограничений среды.

2. Модели представления текста. Тексты, подлежащие анализу, содержат множество явных и неявных признаков, указывающих на различные авторские характеристики: пол, возраст, опыт, идеологию, настроение и др. Каждый такой признак может оказать существенное влияние на конечный результат, поэтому важно представить текст в виде информативных признаков.

3. Алгоритмы разбора текста. Для разбора текста на различных уровнях (лексическом, морфологическом синтаксическом, семантическом) в рамках текстового анализа используются различные методы и инструменты машинного обучения и обработки естественного языка.

4. Методы принятия решений. Ключевой частью методологии является процесс принятия решения о принадлежности текста к классу в зависимости от задачи. Решение принимается статистическими методами, использующими меры расстояния и сходства признаков в пространстве, методами на основе машинного обучения, методами на основе глубокого обучения. Каждая группа методов имеет свои преимущества и недостатки по отношению к конкретной задаче текстового анализа. При этом методы могут применяться как по отдельности, так и совместно в качестве алгоритмического ансамбля.

5. Методы оптимизации гиперпараметров. Большинство задач машинного обучения сводятся к поиску параметров модели, которые минимизируют некоторую функцию потерь. Функция потерь оценивает, насколько хорошо модель соответствует данным.

3 Стремоухов В. Д. Модель и метод анализа схожести и определения авторства вредоносного кода: дис. канд. техн. наук: 05.13.19 / В. Д. Стремоухов. – НИУ ИТМО, Санкт-Петербург, 2013. – 95 с.

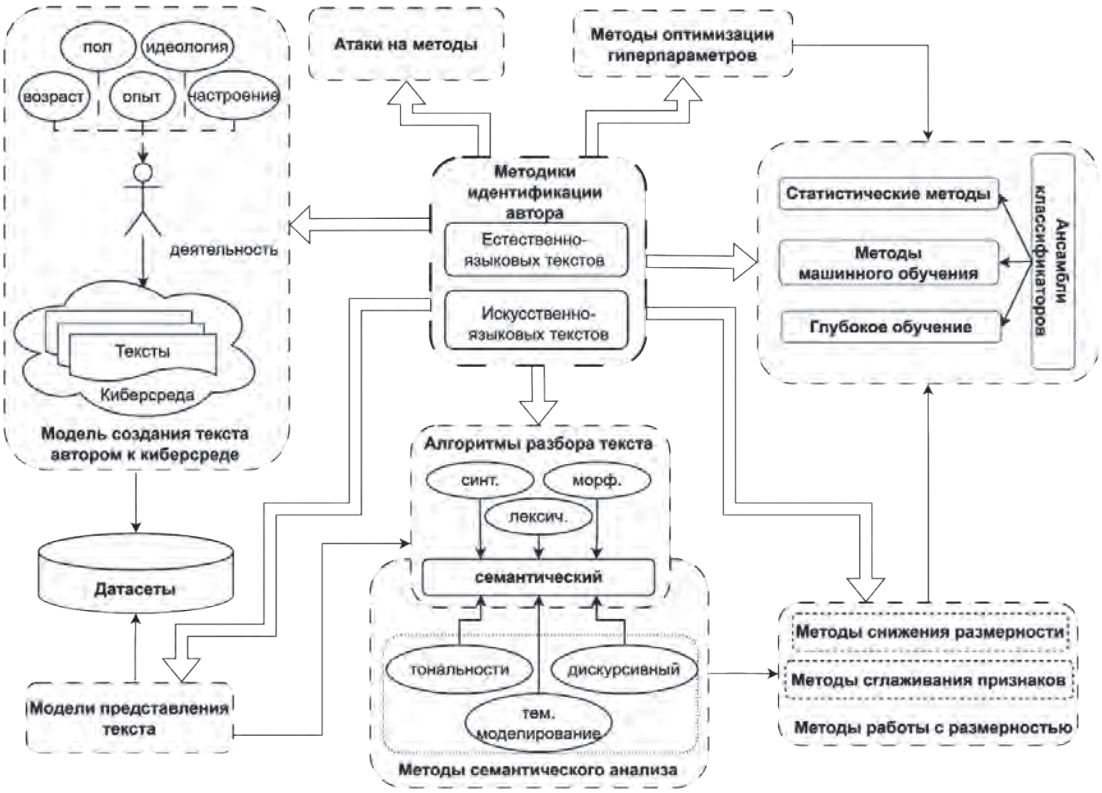


Рис. 1. Методология идентификации автора текстовой информации для решения задач информационной безопасности

Благодаря оптимизации гиперпараметров, обучение становится более эффективным и быстрым, обеспечивается сходимость алгоритма и повышается качество конечной модели.

6. Методы снижения размерности и отбора признаков. Этот процесс помогает повысить эффективность и качество моделей машинного обучения. Во-первых, за счет снижения размерности уменьшается вычислительная сложность. Модели машинного обучения, особенно полученные путем глубокого обучения, могут быть требовательны к вычислительным ресурсам. Уменьшение числа признаков может существенно сократить время обучения и предсказания, а также снизить требования к памяти. Во-вторых, применение информативных признаков в противовес полному множеству признаков позволяет улучшить обобщающую способность модели и предотвратить проблему переобучения за счет удаления нерелевантных или избыточных признаков. В-третьих, в результате снижения размерности признакового пространства повышается интерпретируемость модели: модели с меньшим числом признаков легче интерпретировать.

7. Атаки на методы. Атаки на методы – это специальные методы, предназначенные для запутывания моделей машинного обучения. Среди таких атак можно выделить обфускацию текста, искусственную-генерацию текста и усреднение признаков. Обфускация подразумевает добавление шума в виде опечаток, замену слов синонимами или бессмысленными словами, изменение порядка слов, применение гомоглифов (символов иноязычного алфавита, визуально похожих на исходный), а также использование перефразирования. Искусственная генерация позволяет выполнять имитацию стиля автора за счет использования современных генеративных моделей семейства GPT. С их помощью создаются реалистичные тексты, способны запутать классификатор. Эти атаки могут быть статическими, созданными один раз и используемыми без изменений, или динамическими, адаптирующимися к обновлениям модели классификации. Применяемые для решения задач информационной безопасности модели классификации текстов должны быть устойчивыми к атакам на метод.

8. Алгоритмы сглаживания. Информативные признаки текста, свойственные текстам больших объемов, могут не проявиться в текстах небольших объемов. Алгоритмы сглаживания позволяют оценить вероятности не наступивших событий.

9. Методики текстового анализа. Методология идентификации автора текстовой информации является основой для создания частных методик, позволяющих решать практические задачи информационной безопасности. Методика анализа естественно-языкового текста [16] позволяет решать задачи открытой и закрытой атрибуции автора текста. Закрытая атрибуция является более простым

случаем, т.к. подразумевает наличие истинного автора текста среди авторов-кандидатов. Открытая атрибуция является более сложным случаем ввиду отсутствия истинного автора среди кандидатов. Независимо от вида анализа, методика является устойчивой к атакам на метод и позволяет эффективно идентифицировать автора текста. Методика анализа искусственно-языкового текста [17] позволяет идентифицировать автора исходных кодов программ, учитывая специфику, отличающую их от естественно-языковых текстов, и позволяет осуществлять эффективную идентификацию вне зависимости от языка программирования, квалификации программиста и осложняющих факторов.

Модель создания автором текста в киберсреде с учетом семантики

Модель создания текста автором в киберсреде с учетом семантики представим тройкой:

$$M = (\mathbf{A},\mathbf{T},\mathbf{E}), \quad (1)$$

где $\mathbf{A}$ – множество авторов, $\mathbf{T}$ – множество текстов, $\mathbf{E}$ – множество сред.

Пусть имеется коллекция текстов $\mathbf{T} = \{t_1,...,t_{|T|}\}$ и множество авторов $\mathbf{A} = \{a_1,...,a_{|A|}\}$. Введем бинарное отношение «текст написан автором» $R \subset \mathbf{T} \times \mathbf{A}$ на декартовом произведении множеств $\mathbf{T}$ и $\mathbf{A}$ такое, что выполняется tRa если текст $t \in \mathbf{T}$ автором $a \in \mathbf{A}$:

$$\exists t \in \mathbf{T}, \exists a \in \mathbf{A} : (tRa). \quad (2)$$

В случае, когда текст t можно представить как объединение фрагментов $t = \bigcup_{i=1}^{N_t} t'_i$, написанных несколькими авторами, будем говорить, что текст t «написан в соавторстве»:

$$\exists t'_i, t'_j \subseteq t, \exists a_l, a_m \in \mathbf{A}, a_l \neq a_m : (t'_i Ra_l) \wedge (t'_j Ra_m). \quad (3)$$

Случай, когда текст, написанный одним автором, подвергается изменению другим автором при сохранении общей семантики и тональности текста назовем «редактированием» и опишем в виде функции:

$$t^e = edit(t). \quad (4)$$

Текст, полученный в результате работы генеративного алгоритма, обученного на текстах $\mathbf{T}^a$ автора a, упрощенно опишем в виде:

$$t^g = gen(\mathbf{T}^a). \quad (5)$$

Общий случай вмешательства в процесс создания текста обозначим как $inter \in \mathbf{INTER}$, где элементами множества являются факты самостоятельного написания, редактирования, соавторства, применения генеративного алгоритма.

Текст имеет семантическое описание $topic \in \mathbf{TOPIC}$. Например, такими темами порталов Интернет могут быть «новости», «язык программирования C++», «научные статьи по защите информации» и др. В свою очередь каждую тему можно представить списком ключевых слов (облаком тэгов)

$$topic_i \in \mathbf{TOPIC} = \{keyword_1,...,keyword_{|topic_i|}\}.$$

В контексте решения задач информационной безопасности будем использовать абстрактное понятие «тип текста», которое учитывает вид, стиль, жанр, назначение текста и др. $type \in \mathbf{TYPE}$. Например, в качестве типов в зависимости от задачи будем понимать художественные, любительские и сообщения из социальных сетей, естественно-языковые и искусственно-языковые и т.д.

Текст может быть отнесен к экстремистским материалам. Его действующий статус можно представить как $status \in \mathbf{STATUS}$. Возможные значения признака: разрешен или запрещен.

Каждый текст имеет эмоциональный окрас (тональность) $emo \in \mathbf{EMO}$. Эмоциональный окрас в простом случае может принимать значения: положительный, негативный, нейтральный. Возможна более детальная классификация, учитывающая оттенки радости, злости, грусти, страха, интереса и т.д.

Таким образом множество классов, к которым можно отнести текст можно описать как декартово произведение вышеозначенных множеств, а конкретный класс, к которому относится текст представить как:

$$C^t = (type,status,topic,emo) \in \mathbf{TYPE} \times \mathbf{STATUS} \times \mathbf{TOPIC} \times \mathbf{EMO} \times \mathbf{INTER}. \quad (6)$$

Каждый автор в момент создания текста представляется набором атрибутов, которые можно описать как:

$$\mathbf{C}^a = (id,age,gender,emo,status,action,pop) \in \mathbf{ID}\times\mathbf{AGE}\times\mathbf{GENDER}\times\mathbf{EMO}\times\mathbf{STATUS}\times\mathbf{ACTION}\times\mathbf{POP}, \quad (7)$$

где:

$id \in \mathbf{ID}$ – идентификатор: любая последовательность символов, которой человек себя идентифицирует (ФИО, псевдоним в социальных сетях и др.);

$gender \in \mathbf{GENDER}$ – пол и гендерная идентичность (мужской, женский, представитель ЛГБТ);

$emo \in \mathbf{EMO}$ – настроение, с которым автор писал текст (положительное, отрицательное, нейтральное);

$status \in \mathbf{STATUS}$ – статус, показывающий имеет ли человек статус иностранного агента или экстремиста, или имеет отношение к организациям, имеющим эти статусы;

$action \in \mathbf{ACTION}$ – деятельность автора по созданию текстовой информации (общение, творческая и профессиональная деятельность);

$pop \in \mathbf{POP}$ – категории популярности автора;

$age \in \mathbf{AGE}$ – возрастная группа автора.

Каждый элемент текста описывается вектором признаков, отражающим его свойства. Для естественных текстов у слова, например, можно определить часть речи, морфологические признаки и длину и т.д. Для исходных кодов программ можно определить тип токена, семантику оператора и др. Набор

признаков текста можно представить как результат работы функции извлечения полного вектора характеристик из текста:

$$\mathbf{F} = extract(t) = [f_1,...,f_{|\mathbf{F}|}], \quad (8)$$

где:

- ✓ $f_i \in \mathbf{LEX} \cup \mathbf{MORPH} \cup \mathbf{SYNT} \cup \mathbf{SEM} \cup \mathbf{IDIO} \cup \mathbf{META} \cup \mathbf{EMB}$, LEX – лексические признаки;
- ✓ MORPH – морфологические признаки;
- ✓ SYNT – синтаксические признаки;
- ✓ SEM – семантические признаки;
- ✓ IDIO – идиосинкразические признаки;
- ✓ META – метаданные текста;
- ✓ EMB – некоторое векторное представление текста, полученное с помощью модели машинного обучения, где l – размер входного слоя модели.

Множество информативных характеристик текста $\mathbf{F}^{inf}$ и их значения зависят от типа текста и от атрибутов автора, рассматриваемых в рамках интересующей задачи идентификации. Получение информативных признаков можно представить в виде функции, принимающей на вход текстовые признаки, тип текста и атрибуты автора:

$$\mathbf{F}^{inf} = inf(\mathbf{F}, type, \mathbf{H}) = [f_1^{inf},...,f_{\mathbf{F}^{inf}}^{inf}], \quad (9)$$

Текстами и совокупностью векторов признаков текстов, написанных авторами, все или некоторые атрибуты которых совпадают, можно представить стиль определенной категории авторов:

$$\forall a_l, a_m \in \mathbf{A},\ \mathbf{K}^{a_l=}\ \mathbf{K}^{a_m} = \mathbf{K},\ \mathbf{K} \subseteq \mathbf{C}^{\mathbf{A}}: \mathbf{C}^{\mathbf{K}} = \{\mathbf{F}^{inf_i^{\mathbf{K}}}\}_{i=1}^{N_{\mathbf{T}^{\mathbf{K}}}} = \begin{cases} f^{inf_{1,1}^{\mathbf{K}}},...,f^{inf_{1,|\mathbf{F}^{inf\mathbf{K}}|}^{\mathbf{K}}}) \\ f^{inf_{N_{\mathbf{T}^{\mathbf{K}},1}}^{\mathbf{K}}},...,f^{inf_{N_{\mathbf{T}^{\mathbf{K}},|\mathbf{F}^{inf\mathbf{K}}|}}^{\mathbf{K}}}, \end{cases} \quad (10)$$

где:

- ✓ $\mathbf{K}$ – подмножество совпадающих атрибутов авторов;
- ✓ $\mathbf{T}^{\mathbf{K}}$ – множество текстов, написанных авторами с одинаковыми атрибутами;
- ✓ $N_{\mathrm{T}}^{\mathrm{K}}$ – количество текстов в этом множестве;
- ✓ $\mathbf{F}^{inf_i^{\mathbf{K}}}$ – множество значений лексических, морфологических, синтаксических, семантических, идиосинкразических признаков и эмбеддингов информативных для определенной категории авторов, имеющих одинаковые атрибуты.

Стиль автора может меняться со временем, и значения характеристик $\mathbf{F}$ могут изменяться. Однако множество информативных признаков $\mathbf{F}^{inf}$ должно быть устойчивым к этим изменениям во времени и учитывать небольшое редактирование другими авторами.

Среду, в которой автор пишет или публикует свой текст опишем как:

$$\mathbf{E} = (topic, type, status, rules, pop) \in \mathbf{TOPIC} \times \mathbf{TYPES} \times \mathbf{STATUS} \times \mathbf{RULES} \times \mathbf{POP}, \quad (11)$$

где:

- ✓ $topic \in \mathbf{TOPIC}$ – семантическое описание среды, т.е. список тематик, тексты, относящиеся к которым, размещаются в среде;
- ✓ $type \in \mathbf{TYPES}$ – тип среды. Например, Интернет-библиотека, мессенджер, социальная сеть, хостинг IT-проектов, Интернет-СМИ, сайт научного журнала, локальный компьютер автора и др.;
- ✓ $status \in \mathbf{STATUS}$ – действующий статус ресурса (разрешен или запрещен). $rules \in \mathbf{RULES}$ – правила размещения текста в среде и/или стандарты среды и необходимо ли им следовать. Например, к ним можно отнести соблюдение законодательства РФ, правил публикации статей в научном журнале, стандартов кодирования для исходных текстов программ на хостинге IT-проектов в компании, запрет публикации текстов, не соответствующих определенной возрастной категории и др.;
- ✓ $pop \in \mathbf{POP}$ – популярность среды, каждый тип среды имеет свою специфику оценки. Например, для Интернет-ресурса – количество посетителей, посетивших его за определенный период времени; для сообщества в социальной сети или канала в мессенджере – количество реальных подписчиков; сайт научного журнала – импакт-фактор и т.д.

Отметим, что чем популярнее среда, тем больший охват аудитории она имеет и тем быстрее опубликованная текстовая информация находит читателя.

Введем бинарное отношение «текст создается в среде» $Q \subset \mathbf{T} \times \mathbf{E}$ на декартовом произведении множеств $\mathbf{T}$ и $\mathbf{E}$ такое, что выполняется tQe, если текст $t \in \mathbf{T}$ создается и размещается в среде $e \in \mathbf{E}$:

$$\exists t \in \mathbf{T}, \exists e \in \mathbf{E}: (tQe). \quad (12)$$

Решение задач информационной безопасности (рис. 2) сводится к отнесению текста t_{K} к классу $c \in \mathbf{C} = \mathbf{C}^t \cup \mathbf{C}^{\mathbf{K}}$ с учетом ограничений и особенностей среды $\mathbf{E}$ на основе текстов, класс для которых известен $\mathbf{T}' = \{t_1,...,t_m\} \subseteq \mathbf{T}$, т.е. существует множество пар «текст-класс» $\mathbf{D} = \{t_i, c_j\}_{i=1}^{m}$. Целью является построение классификатора, решающего данную задачу, т.е. нахождение некоторой целевой функции $\Phi: \mathbf{T} \times \mathbf{C} \to [0,1]$. Значения функции интерпретируется как степень принадлежности объекта классу: 1 соответствует полностью положительному решению, 0 – отрицатель-

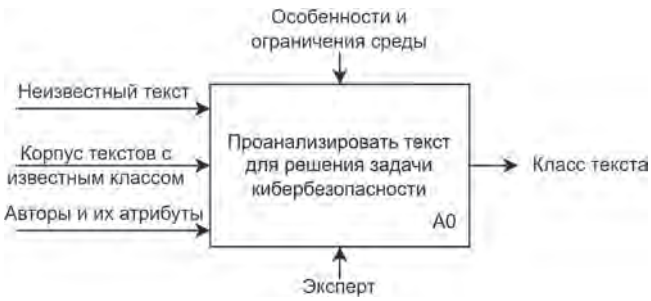


Рис. 2. IDEF0 диаграмма процесса анализа текста для решения задач кибербезопасности

Таблица 1

Примеры ограничений и влияния атрибутов автора, текста и среды

Атрибут текста	Атрибут автора	Атрибут среды	Ограничения и влияние атрибутов
t.topic	a.age	e.rules	Возрастные ограничения среды ограничивают тематику текстов, которые может публиковать или читать автор определенного возраста.
t.type	a.gender	e.topic	Некоторые среды более популярны среди определенного пола, что может влиять на выбор жанра или типа текста автором.
t.emo	a.action	e.type	Профессиональная деятельность автора коррелирует с выбором профессиональных сред и специализированных тем, что делает тексты более строгими по тональности.
t.status	a.status	e.status	Статус автора как иноагента и статус среды как разрешенного или запрещенного ресурса напрямую влияют на легальность и доступность текста для чтения.
t.status	a.action	e.rules	Авторы, занимающиеся творческой деятельностью, могут сталкиваться с ограничениями в средах со строгими правилами относительно контента.
t.topic	a.topic	e.pop	Популярные авторы выбирают тематики текстов, соответствующие интересам большинства пользователей популярной среды, чтобы увеличить свое влияние и расширить аудиторию.
t.type	a.action	e.type	Авторы-ученые часто публикуют научные работы в специализированных журналах, где тип среды соответствует типу текста.
t.status	a.status	e.rules	Авторы текстов, признанных иноагентами, могут сталкиваться с дополнительными ограничениями в определенных средах из-за строгих правил.

ному. При этом каждый текст рассматривается как вектор признаков $\mathbf{F}$.

Примеры ограничений среды, накладывающихся на авторов и создаваемые тексты, а также влияние атрибутов друг на друга приведены в таблице 1.

Интерпретация в виде графа представлена на рис. 3.

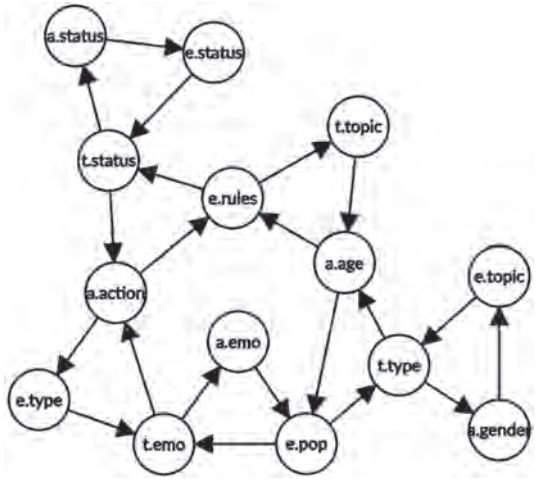


Рис. 3. Ограничения и влияние атрибутов автора, текста и среды

Заключение

В статье предлагается комплексное решение важной научной проблемы идентификации автора текстовой информации и таких авторских атрибутов, как пол и гендер, возраст, идеология и взгляды, основанное на передовых технологиях искусственного интеллекта и машинного обучения.

1. На основе анализа материалов предложена методология идентификации автора естественно-языкового текста и исходных текстов программ для решения задач информационной безопасности в виде комплекса методов, моделей и алгоритмов, агрегирующий имеющийся опыт. Методология является универсальной для решения задач информационной безопасности, связанных с классификацией текстов.

2. Предложена модель создания автором текста в киберсреде, учитывающая взаимодействие компонентов и ограничения, накладываемые на процесс личностью автора и видом деятельности, особенностями среды, текста и семантики.

3. Методология предполагает использование методов статистического анализа, машинного и глубокого обучения. Исходя из специфики текстов, решаемой задачи информационной безопасности и потенциальных атак на методы, в элементах методологии могут быть задействованы разные по принципу действия подходы. Традиционные методы машинного обучения обеспечивают высокую степень интерпретируемости и скорости, поэтому могут применяться как самостоятельно, так и в составе ансамблей методов принятия решений, однако являются менее эффективными в сравнении с НС при наличии осложняющих факторов. НС являются более устойчивыми к атакам на метод и более эффективными для поиска явных и неявных признаков авторского стиля.

4. Ключевыми в методологии являются методики идентификации автора естественно-язычного и искусственно-язычного текста, так как подразумевают

применение разных по своему принципу действия методов и инструментов. Это связано со спецификой анализируемых данных. Подходы, используемые для естественно-языкового анализа, должны обеспечивать интерпретацию неоднозначности, понимание контекста, а также разрешение двусмысленности, которые являются менее характерными для искусственно-языковых текстов. Подходы, используемые для искусственного языка, напротив, должны быть адаптированы под анализ регулярных структур, информативные признаки в которых могут быть менее выраженными ввиду следования авторами строгим синтаксическим и семантическим инструкциям и правилам.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках базовой части государственного задания ТУСУРа на 2023–2025 гг. (проект № FEWM-2023-0015).

Литература

1. *Uslu U., Durmaz Ö., Alptekin G. I. Evaluation of Deep Learning Models for Continuous Authentication Using Behavioral Biometrics // Proceedings of 27th International Conference on Knowledge Based and Intelligent Information and Engineering Sytems (KES 2023), Procedia Computer Science. – 2023. – Vol. 225. – P. 1272–1281.*
2. *Bano H, Akbar W., Aslam N., Bilal M. Identification and Classification of Extremist by Topic Modeling Sentiment Analysis // VFAST Transactions on Software Engineering. – 2023. – Vol. 11. – P. 235–248.*
3. *Аванесян Н. Л., Соловьев Ф. Н., Тихомирова Е. А., Чеповский А. М. Выявление значимых признаков противоправных текстов // Вопросы кибербезопасности. – 2020. – № 4(38). – С. 76–84.*
4. *Васильев В. И., Вульфин А. М., Кучкарова Н. В. Тематическое моделирование и суммаризация текстов в области кибербезопасности // Вопросы кибербезопасности. – 2023. – № 2(54). – С. 1–22*
5. *Araque O., Iglesias C. A. An Approach for Radicalization Detection Based on Emotion Signals and Semantic Similarity // IEEE Access. – 2020. – Vol. 8. – P. 17877–17891.*
6. *Asad M., Shafiq Z., Srinivasan P. A Girl Has A Name: Detecting Authorship Obfuscation // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics. – 2020. – P. 2235–2245.*
7. *Kovalev A. K., Kuznetsova Yu. M. Possibilities of automatic text analysis in the task of determining the psychological characteristics of the author // Experimental Psychology (Russia). – 2020. – Vol. 13, no. 1. – P. 149–158.*
8. *Москин Н. Д. Теоретико-графовые модели, методы и программные средства интеллектуального анализа текстовой информации на примере фольклорных и литературных произведений: дис. д-р. техн. наук: 05.13.18. – Петрозаводский. гос. университет, Петрозаводск, 2022. – 346 с.*
9. *Рогов А. А. Проблема атрибуции в журналах «Время», «Эпоха» и еженедельнике «Гражданин» / А. А. Рогов, Р. В. Абрамов, Д. Д. Бучнева, О. В. Захарова, К. А. Кулаков, А. А. Лебедев и др. // Издательство «Острова». – 2021. – 391 с.*
10. *Огорелков И. В. Исследование лингвистических характеристик текста с целью определения пола автора на примере анализа письменных русскоязычных текстов политического дискурса: дис. канд. техн. наук: 10.02.01. – ФГБОУ ВО «Государственный институт русского языка им. А.С. Пушкина», Москва, 2021. – 457 с.*
11. *Огорелков И. В. Исследование лингвистических характеристик письменного текста политического дискурса с целью определения пола автора // Язык. Право. Общество: сб. ст. V Междунар. науч.-практ. конф. (г. Пенза, 22–25 мая 2018 г.) / под общ. ред. О. В. Барабаш; редколегия: М. Б. Ворошилова, Т. В. Дубровская, А. К. Дятлова, Н. А. Павлова. – Пенза: Изд-во ПГУ, 2018. – 484 с. ISBN 978-5-907018-83-9. – 2018. – С. 88–93.*
12. *Сбоев А. Г. Нейросетевое моделирование и машинное обучение на основе экспериментальных и наблюдательных данных: дис. д-р. техн. наук: 05.13.18. – Национальный исследовательский центр «Курчатовский институт», Москва, 2021. – 389 с.*
13. *Sboev A. Neural Network Model to Include Textual Dependency Tree Structure in Gender Classification of Russian Text Author / A. Sboev, A. Selivanov, R. Rybka, I. Moloshnikov, D. Bogachev // Advanced Technologies in Robotics and Intelligent Systems. – Springer, Cham, 2020. – P. 405–412.*
14. *Давыдова Ю. В. Методы текстового поиска и обработки информации в социальных сетях при управлении деятельностью правоохранительных органов: дис. канд. техн. наук, 05.13.10. – ФГБОУ ВО «Орловский государственный университет имени И. С. Тургенева», Белгород, 2021. – 146 с.*
15. *Андреев И. А. Исследование методов и алгоритмов обработки текстовой информации социальных сетей в задачах формирования социального портрета пользователя: дис. канд. техн. наук, 05.13.01. – Ульяновский государственный технический университет, Ульяновск, 2022. – 166 с.*
16. *Куртукова А. В., Романов А. С., Федотова А. М., Шелупанов А. А. Применение методов машинного обучения и отбора признаков на основе генетического алгоритма в решении задачи определения автора русскоязычного текста для кибербезопасности / А. В. Куртукова [и др.] // Доклады ТУСУР. – 2022. – Т. 25, № 1. – С. 79–85.*
17. *Романов А. С., Куртукова А. В., Шелупанов А. А., Федотова А. М. Идентификация автора исходного кода программы на основе неоднородных данных для решения задач кибербезопасности / А. В. Куртукова, А. А. Шелупанов, А. М. Федотова // Моделирование, оптимизация и информационные технологии. – 2022. – №10(3) [Электронный ресурс]. – URL: https://moitvivt.ru/ru/journal/pdf?id=1227 DOI: 10.26102/2310-6018/2022.38.3.016.*

МЕТОД ОБНАРУЖЕНИЯ ПОДОЗРИТЕЛЬНЫХ ТРАНЗАКЦИЙ БАНКОВСКИХ КЛИЕНТОВ НА ОСНОВЕ СИСТЕМЫ РАСПОЗНАВАНИЯ ЭМОЦИЙ

Козьминых С. И.[1], Татаренков В. С.[2]

DOI: 10.21681/2311-3456-2024-3-129-140

Цель статьи: *разработка метода выявления транзакций, совершенных клиентами, подвергнутыми воздействию мошенников с помощью методов социальной инженерии, на основе анализа видеоданных лица с использованием нейросетевых методов распознавания эмоций.*

Метод исследования: *анализ современных нейросетевых моделей и подходов, используемых для решения задачи распознавания эмоций; анализ архитектур нейронных сетей, обрабатывающих видеоизображение или последовательность кадров; разработка и программная реализация метода обнаружения подозрительных транзакций с использованием искусственных нейронных сетей по видеоданным лица человека; экспериментальное исследование и оценка разработанного метода.*

Полученный результат: *разработан метод выявления подозрительных транзакций, основанный на нейросетевых методах распознавания лицевых эмоций клиентов банка, подвергшихся воздействию злоумышленников. Реализована комбинированная структура нейронной сети с использованием архитектуры пригодной для обработки графической информации и информации, представленной во временной последовательности, для решения задачи распознавания эмоций. Создан программный прототип, позволяющий оценивать эмоциональное состояние наблюдаемого человека по видеоданным лица и способный определять нахождение человека в негативном эмоциональном состоянии. Были проанализированы результаты разработанного метода. Даны рекомендации по перспективам его применения и дальнейшим исследованиям данной темы.*

Научная новизна: *предложен новый метод выявления подозрительных транзакций, основанный на решении задачи распознавания эмоций по видео с применением комбинации CNN и LSTM архитектур нейронных сетей.*

Ключевые слова: *долгая краткосрочная память, машинное обучение, распознавание эмоций, рекурентные нейронные сети, сверточные нейронные сети, CNN, LSTM.*

METHOD FOR DETECTING SUSPICIOUS TRANSACTIONS OF BANKING CLIENTS BASED ON EMOTIONRECOGNITION SYSTEM

Kozminykh S. I.[3], Tatarenkov V. S.[4]

The purpose of the article: *to develop a method for detecting transactions made by customers exposed to fraud using social engineering methods based on the analysis of video data of a person using neural network methods of emotion recognition.*

1 Козьминых Сергей Игоревич, доктор технических наук, доцент, профессор кафедры информационной безопасности Финансового университета при Правительстве РФ, профессор кафедры прикладной информатики и информационной безопасности РЭУ им. Г. В. Плеханова, г. Москва, Россия. E-mail: SIKozminykh@fa.ru, Kozminyh.SI@rea.ru

2 Татаренков Владислав Сергеевич, аспирант кафедры информационной безопасности Финансового университета при Правительстве РФ г. Москва, Россия. E-mail: vt96@mail.ru

3 Sergey I. Kozminykh, Doctor of Technical Sciences, Associate Professor, Professor of the Department of Information Security of the Financial University under the Government of the Russian Federation, Professor of the Department of Applied Informatics and Information Security of Plekhanov Russian University of Economics, Moscow, Russia. E-mail: SIKozminykh@fa.ru, Kozminyh.SI@rea.ru

4 Vladislav S. Tatarenkov, postgraduate student of the Department of Information Security of the Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: vt96@mail.ru

Research method: *analysis of modern neural network models and approaches used to solve the problem of emotion recognition; analysis of neural network architectures that process a video image or sequence of frames; development and software implementation of a method for detecting suspicious transactions using artificial neural networks based on video data of a person's face; experimental research and evaluation of the developed method.*

The result obtained: *a method for detecting suspicious transactions based on neural network methods for recognizing the facial emotions of bank customers exposed to intruders has been developed. A combined neural network structure is implemented using an architecture suitable for processing graphical information and information presented in a time sequence to solve the problem of emotion recognition. A software prototype has been created that allows you to assess the emotional state of an observed person from video data of a person and is able to determine whether a person is in a negative emotional state. The results of the developed method were analyzed. Recommendations are given on the prospects of its application and further research on this topic.*

Scientific novelty: *a new method for detecting suspicious transactions is proposed, based on solving the problem of recognizing emotions from video using a combination of CNN and LSTM architectures of neural networks.*

Keywords: *long-term short-term memory, machine learning, emotion recognition, recurrent neural networks, convolutional neural networks, CNN, LSTM.*

Введение

На сегодняшний день, согласно опубликованным отчетам об исследованиях рынка систем обнаружения и распознавания эмоций, данные системы находят широкое применение в различных сферах деятельности человека[5]. Современные системы распознавания эмоций способны не только классифицировать стандартные типы эмоций, такие как радость, грусть, злость и так далее, но и определять специфические эмоции.

Вначале рассмотрим одну из таких сфер, которой является медицина и здравоохранение, где подобного рода системы призваны определять состояние людей по визуальному наблюдению с возможностью дополнительного анализа показателей, поступающих с медицинских измерительных приборов [1,2]. Также в опубликованных работах встречается описание применения систем распознавания эмоций для диагностики стрессового и психического расстройств [3,4]. Следующей рассматриваемой сферой является сфера автотранспорта и дальних грузоперевозок, для которой ведутся разработки и исследования систем по определению утомления или стрессового состояния человека [5,6]. Основной задачей, решаемой данными системами, является заблаговременное определение критического состояния водителя по камерам из салона машины для предотвращения возникновения потенциальных дорожно-транспортных происшествий. Данные примеры исследований и разработок показывают, что современные методы для распознавания эмоций способны использоваться для определения психоэмоционального состояния человека и сложных глубинных эмоций.

Рассмотрим банковскую сферу. Поиск информации о внедрении систем обнаружения и распознавания эмоций в банках дает результаты, где эти системы применяются в основном в маркетинговых целях[6]. В частности, это применяется для определения удовлетворения клиента качеством обслуживания или определения эмоционального состояния клиента для последующей рекомендации ему того или иного банковского продукта. В свою очередь, по статистике, ежегодно публикуемой компаниями регуляторами[7] в сфере информационной безопасности можно сделать вывод о стабильно высоком уровне телефонного и СМС-мошенничества, направленного

5 1. Emotion Detection and Recognition Market, By Technology Type (Facial Expression Recognition, Speech Emotion Recognition, and Biometric Recognition), By Component (Hardware, Software, and Services), By Application, and By Region Forecast to 2032 // Reports and Data : сайт. – 2023. – URL: https://www.reportsanddata.com/report-detail/emotion-detection-and-recognition-market (дата обращения: 01.12.2023).
2. Ankit Gupta. Emotion Detection and Recognition market Research Report Information By Technology (Bio Sensors Technology, Machine Learning, Pattern Recognition, Feature Extraction and 3D Modelling, Natural Language Processing (NLP), Others), By Service (Storage and Maintenance, Consulting and Integration), By Application (Law Enforcement, Surveillance and Monitoring, Marketing & Advertising, Media & Entertainment, Others), And By Region (North America, Europe, Asia-Pacific, And Rest Of The World) – Market Forecast Till 2030 // Market Research Future : сайт. – 2024. – URL: https://www.marketresearchfuture.com/reports/emotion-detection-recognition-market-3193 (дата обращения: 19.01.2024).

6 1. Будущее за эмоциями: как Альфа-Банк улучшает клиентский опыт // Forbes : новостной портал. – 2023. – URL: https://www.forbes.ru/brandvoice/501354-budusee-za-emociami-kak-al-fa-bank-ulucsaet-klientskij-opyt (дата обращения: 04.12.2023).
2.Нейросети банка «Точка» научились распознавать эмоции клиентов // Adindex : новостной портал. – 2023. – URL: https://adindex.ru/news/digital/2023/06/1/312986.phtml (дата обращения: 05.12.2023).

7 1. Kaspersky Who Calls: в первом квартале 2023 года доля столкнувшихся с классическим телефонным мошенничеством выросла на три процентных пункта // Kaspersky : сайт. – 2023. – URL: https://www.kaspersky.ru/about/press-releases/2023_kaspersky-who-calls-v-pervom-kvartale-2023-goda-dolya-stolknuvshihsya-s-klassicheskim-telefonnym-moshennichestvom-vyrosla-na-tri-procentnyh-punkta (дата обращения: 06.12.2023).
2. Инциденты информационной безопасности: итоги I квартала 2023 года // CBR : сайт. – 2023. – URL: https://cbr.ru/press/event/?id=15814 (дата обращения: 07.12.2023).
3. Рост количества мошеннических операций за 2023 год — статистика от ЦБ РФ // Finadvice MTS : сайт. – 2023. – URL: https://finadvice.mts.ru/blog/rost-kolichestva-moshennicheskih-operatsii-za-2023-god--statistika-ot-tsb-rf (дата обращения: 08.12.2023).

на запугивание жертвы и доведение его до стрессового состояния, в котором жертва произведет денежный перевод на счет мошенников. Переводы в большинстве таких случаев осуществляются через мобильное приложение или через банкомат, а именно, через устройства, в которых есть камера для получения и анализа видеосигнала. Таким образом имеется возможность отследить эмоциональное состояние клиента и пометить его транзакции, в случае, если они были совершены в стрессовом состоянии.

В данной статье рассмотрен метод определения подозрительных банковских транзакций с использованием нейросетевых технологий по распознаванию эмоций в период стрессового состояния клиента. Приведен анализ современных подходов и моделей нейронных сетей, используемых для решения задачи распознавания эмоций по видеоизображению. Дано описание используемого набора данных для обучения реализуемой модели, алгоритма предобработки данных и итоговой архитектуры модели на основе нейросетей с CNN и LSTM архитектурами.

1. Анализ архитектур нейронных сетей и методов машинного обучения, используемых в системах распознавания эмоций и обработки временных последовательностей данных

Для реализации предложенного в данной статье метода анализировались публикации, в которых применялись нейросетевые архитектуры и алгоритмы машинного обучения для обработки графической информации, представленной изображениями, последовательностями кадров или видео. А также статьи в области решения задачи по распознаванию эмоций и обработки последовательностей данных для выявления в них хронологических зависимостей.

В статье [7] приведен обзор современных методик и подходов к моно- или мультимодальному распознаванию эмоций, в том числе и по лицевой информации. В качестве традиционных методов извлечения лицевых признаков упоминаются следующие источники: Local Binary Pattern (LBP), Active Appearance Model (AAM), Active Shape Model (ASM), Histograms of Oriented Gradient (HOG), Gabor Wavelet Transform. К недостаткам традиционных методов относят необходимость ручного вмешательства в подготовку данных, а также неспособность извлечь и сохранить всю семантическую информацию лица, необходимую для определения эмоций. Для нивелирования данных особенностей традиционных методов предлагается использование сверточных нейронных сетей (CNN). Для работы с последовательностью изображений указывается применение архитектуры трехмерной сверточной нейронной сети 3D-CNN.

В публикации [8] приведен литературный обзор статей по теме распознавания эмоций по лицу с указанием используемых методов, наборов данных и технологий. В большинстве работ выделяют 7 базовых эмоций, которые учитываются при обучении и тестировании построенных моделей: злость (anger), грусть (sadness), радость (happiness), отвращение (disgust), удивление (surprise), страх (fear) и нейтральное состояние (neutral). Для данных классов эмоций применяют группировку по двум категориям: позитивные эмоции (удивление, радость) и негативные эмоции (злость, грусть, отвращение, страх). Из приведенных в работе наборов данных, для дальнейшего поиска источников по получению к ним доступа были отобраны наборы CK+ и AFEW, как наиболее подходящие по описанию содержимого для решения поставленной задачи. В качестве широко используемого метода для извлечения лицевых признаков указывается нейросетевая архитектура сверточных нейронных сетей. Для определения лица на изображении используются также обученные нейронные сети или традиционный метод Виолы-Джонса.

В статьях [9–13] для выделения лицевых признаков при обработке изображений лица также используются вариации архитектуры сверточной нейронной сети.

В работе [14] используется комбинированная архитектура нейронной сети, состоящая из частей сверточной нейронной сети и блоков долгой кратковременной памяти. Такая архитектура применяется для обработки последовательности данных, которая в работе представлена речевой записью. В публикации [15] также применяется комбинация сверточной и рекуррентной нейросетевых архитектур для обработки последовательности данных, представленных измерениями фотоплетизмографа. В статье [16] комбинация архитектур CNN и LSTM используется в качестве подхода для классификации действий человека по активностям на основе видеоданных. Как отмечают авторы, данный подход к распознаванию по видео показывает свою эффективность.

Статьи [17,18] демонстрируют возможность размещения и осуществления работы на мобильных устройствах и встраиваемых системах, построенных на базе глубокого машинного обучения.

Анализ публикаций показал, что самым популярным методом для извлечения признаков из графической информации является метод машинного обучения на основе сверточной нейронной сети (CNN). Данный метод решает недостатки традиционных подходов, а также обладает большим количеством изученных и исследованных вариаций моделей. Сверточная нейронная сеть может быть представлена адаптированной моделью из 3-х слоев, моделью по типу VGG-16 из 16-ти слоев, моделью по типу GoogLeNet из 22-х слоев и т.д. Рекуррентные нейронные сети (RNN) широко применяются для анализа

данных, представленных какими-либо временными последовательностями (последовательность измерений, звуковая запись, видео) и извлечения из них признаков, формирующих контекст и зависимость между элементами этой последовательности. Из архитектур рекуррентных нейронных сетей исследователи выделяют модель ячеек долгой краткосрочной памяти (LSTM). Ее преимуществами над традиционными моделями RNN являются: преодоление проблемы исчезающего градиента, способность запоминать информацию и учитывать ее на протяжении множества шагов, работа с последовательностями переменной длины. У LSTM модели также есть усовершенствованная модель Bi-LSTM, которая имеет возможность обрабатывать информацию от прошлых значений к будущим, а также в обратном порядке от будущих значений к прошлым.

Исследователи в задачах распознавания глубоких эмоций (усталость, стресс и т.д.) используют как классические наборы данных с базовыми эмоциями, вручную группируя их по признакам в рамках своей задачи, так и наборы данных, собранные с учетом специфики распознаваемых видов эмоций. В качестве популярных метрик оценки качества разрабатываемых систем для решения задач много классовой классификации применяется метрика accuracy и матрица ошибок (confusion matrix).

2. Разработка системы распознавания стрессового эмоционального состояния клиента банка по видеоданным

Для реализации системы распознавания стрессового эмоционального состояния по лицевым данным, полученным из видео, использовался следующий инструментарий: язык программирования Python, библиотека алгоритмов компьютерного зрения OpenCV, библиотека машинного обучения PyTorch [19], среда разработки PyCharm и интерактивная облачная среда для разработки и выполнения кода Google Colab [20].

2.1. Используемые наборы данных (dataset-ы)

Для обучения и тестирования построенной нейронной сети необходимо использовать наборы коротких видео с людьми, испытывающими эмоции, распределенные по категориям: нормальное состояние (Normal), слабое стрессовое состояние (Weak stress) и сильное стрессовое состояние (Strong stress) [21].

В качестве основного обрабатываемого набора данных использовался DFEW [22], являющийся похожим на набор AFEW, но имеющий упрощенный способ получения доступа к его данным. DFEW составлен на основе 15 906 видеоклипов. Из каждого такого видеоклипа извлекались кадры, на которых определялось лицо, после чего кадр обрезался в соответствии с прямоугольной зоной найденного лица. В свою очередь, видеоклипы прошли процесс классификации по 7 базовым эмоциям, которую осуществляли эксперты. После проведения такой обработки данных пользователям-исследователям набора данных предлагается использовать 11 697 объектов, размеченных по 7 классам (Happy, Sad, Neutral, Angry, Surprise, Disgust, Fear), где каждый объект представлен набором-последовательностью изображений извлеченного лица из видеоклипа.

Таблица 1

Параметры набора данных DFEW

№	Параметр	Значение
1	Количество объектов общее, шт.	11 697
2	Количество объектов класса Happy, шт.	2445
3	Количество объектов класса Sad, шт.	1894
4	Количество объектов класса Neutral, шт.	2669
5	Количество объектов класса Angry, шт.	2173
6	Количество объектов класса Surprise, шт.	1469
7	Количество объектов класса Disgust, шт.	145
8	Количество объектов класса Fear, шт.	902
9	Максимальная высота кадра, пикс.	256
10	Минимальная высота кадра, пикс.	256
11	Максимальная ширина кадра, пикс.	256
12	Минимальная ширина кадра, пикс.	256
13	Максимальное количество кадров у объекта, шт.	847
14	Среднее количество кадров у объекта, шт.	70
15	Минимальное количество кадров у объекта, шт.	20
16	Цветовая модель изображений	RGB

После адаптации данных для обучения и тестирования разрабатываемой нейронной сети, итоговый набор выглядит таким образом, как представлено на рис. 1.

Применимо к данному набору использовался способ с распределением базовых классов эмоций по целевым категориям. Исходя из круговой модели классификации эмоций Рассела [23], было решено распределить данные базовых эмоций следующим образом: Normal (Happy, Neutral), Weak stress (Sad, Surprise, Disgust) и Strong stress (Angry, Fear), представлено на рис. 2.

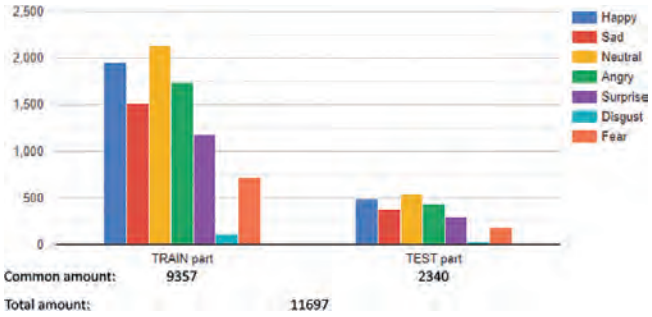


Рис. 1. Распределение данных по классам базовых эмоций по обучающей и тестирующей выборкам полученного набора из DFEW с классификацией 7 базовых эмоций

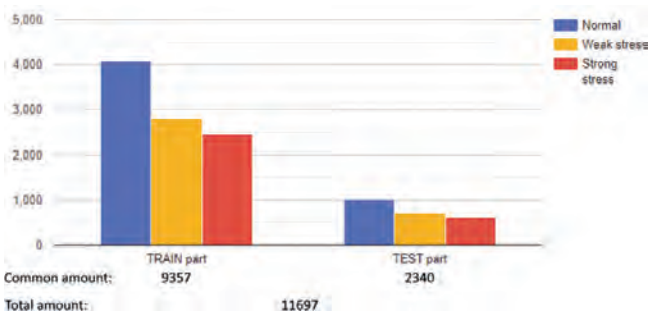


Рис. 2. Распределение данных по целевым категориям для распознавания после группировки базовых эмоций

2.2. Модель нейронной сети

Предлагаемая модель нейронной сети 3DCNN-LSTM построена с применением комбинации сверточных нейронных сетей (CNN) и долгой-краткосрочной памяти (LSTM). Данная архитектура позволяет извлекать признаки из последовательности графической информации, а потом выявлять в них скрытые связи и закономерности. Графическое представление архитектуры реализуемой модели представлено на рис. 3.

На вход нейронной сети подается последовательность из 20 изображений размером 128 x 128 пикселей с 3 цветовыми каналами. Параметр входного количества изображений был выбран исходя из минимального значения количества кадров для объекта во всем наборе данных. Размер подаваемого на вход изображения был подобран эвристическим путем. Во-первых, чтобы обеспечить обработку изображения в хорошем качестве, где лицо на изображении отображается без эффекта пиксельности. Во-вторых, чтобы обеспечить количество слоев сверточной нейронной сети, каждый из которых заканчивается подвыборкой, уменьшающей изображение вдвое. А также для сведения изображения до размеров, позволяющих преобразовать (flatten) его в вектор той размерности, которая требуется, чтобы у LSTM блока было количество обучаемых параметров, умещающееся в оперативную память при обучении сети, и кроме того, не увеличивающее обучение этой сети до нецелесообразно долгих периодов времени.

Одновременно с предыдущим условием, необходимо, чтобы при выбранном количестве слоев не наблюдалось эффекта затухания градиента. Количество каналов было бы выбрано из учета особенности обработки кадров в наборе данных. При извлечении изображения лица из кадров видеоданных, после обнаружения его области, необходимо чтобы данное лицо могло быть выровнено путем применения аффинных преобразований, что приводит к появлению пустых областей в кадре, закрашенных черным цветом, как изображено на рис. 5. При выборе одного канала (оттенки серого), данная область могла внести помехи в процесс обучения, поэтому выбор был сделан в пользу 3 каналов (RGB).

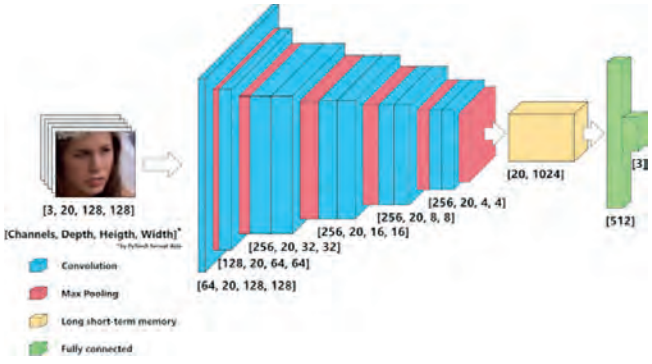


Рис. 3. Графическое представление архитектуры реализуемой модели

```
==========================================================================================
Layer (type:depth-idx)                   Output Shape              Param #
==========================================================================================
CNN3D_LSTM_Net                           [8, 3]                    --
├─Conv3d: 1-1                            [8, 64, 20, 128, 128]     1,792
├─MaxPool3d: 1-2                         [8, 64, 20, 64, 64]       --
├─Dropout: 1-3                           [8, 64, 20, 64, 64]       --
├─Conv3d: 1-4                            [8, 128, 20, 64, 64]      73,856
├─MaxPool3d: 1-5                         [8, 128, 20, 32, 32]      --
├─Dropout: 1-6                           [8, 128, 20, 32, 32]      --
├─Conv3d: 1-7                            [8, 256, 20, 32, 32]      295,168
├─Conv3d: 1-8                            [8, 256, 20, 32, 32]      590,080
├─MaxPool3d: 1-9                         [8, 256, 20, 16, 16]      --
├─Dropout: 1-10                          [8, 256, 20, 16, 16]      --
├─Conv3d: 1-11                           [8, 256, 20, 16, 16]      590,080
├─Conv3d: 1-12                           [8, 256, 20, 16, 16]      590,080
├─MaxPool3d: 1-13                        [8, 256, 20, 8, 8]        --
├─Dropout: 1-14                          [8, 256, 20, 8, 8]        --
├─Conv3d: 1-15                           [8, 256, 20, 8, 8]        590,080
├─Conv3d: 1-16                           [8, 256, 20, 8, 8]        590,080
├─MaxPool3d: 1-17                        [8, 256, 20, 4, 4]        --
├─Dropout: 1-18                          [8, 256, 20, 4, 4]        --
├─Conv3d: 1-19                           [8, 256, 20, 4, 4]        590,080
├─Conv3d: 1-20                           [8, 256, 20, 4, 4]        590,080
├─MaxPool3d: 1-21                        [8, 256, 20, 2, 2]        --
├─Flatten: 1-22                          [8, 20, 1024]             --
├─LSTM: 1-23                             [8, 20, 512]              3,149,824
├─Linear: 1-24                           [8, 512]                  262,656
├─Dropout: 1-25                          [8, 512]                  --
├─Linear: 1-26                           [8, 3]                    1,539
==========================================================================================
Total params: 7,915,395
Trainable params: 7,915,395
```

Рис. 4. Подробное представление модели с помощью встроенной функции библиотеки PyTorch summary()

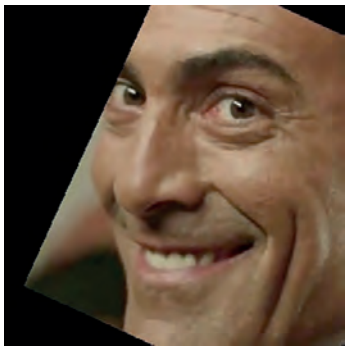


Рис. 5. Пример области пустых пикселей, закрашенных черным цветом

На первом этапе выделения признаков из объектов данных последовательность кадров проходит обработку через структуру 3DCNN сверточной нейронной сети. В ходе такой обработки извлекаются пространственные признаки каждого изображения в последовательности, которые затем подаются на вход структуре долгой-краткосрочной памяти LSTM. Данная структура анализирует и выявляет основные временные признаки последовательности. На последнем этапе данные из LSTM блока подаются на вход блока, состоящего из полносвязных слоев. Последний полносвязный слой содержит 3 нейрона, выход из которых показывает принадлежность классу.

2.2.1. Сверточный слой (Convolutional layer)

В сверточном слое к данным, представленным n-мерной структурой, применяются n-мерные ядра свертки с заданным шагом движения по исходной структуре данных. После прохождения всех ядер по данным формируется выходная карта признаков. В общем случае, математическая формула 3Д сверточного слоя, выглядит следующим образом. Для входной структуры данных X формата (C_{in}, D_{in}, H_{in}, W_{in}), где C_{in} – количество каналов, D_{in} – глубина, H_{in} – высота, W_{in} – ширина. С ядром (фильтром) W размерности (C_{in}, D_{kernel}, H_{kernel}, W_{kernel}):

$$Y_{i,j,k} = \sum_{l=1}^{C_{in}} \sum_{m=1}^{D_{kernel}} \sum_{n=1}^{H_{kernel}} \sum_{p=1}^{W_{kernel}} X_{l,i+m-1,j+n-1,k+p-1} * \\ * W_{l,m,n,p} + bias \quad (1)$$

Эта формула выполняется независимо для каждого выходного канала Cout. Размеры выходной структуры данных (C_{out}, H_{out}, W_{out}) рассчитываются с учетом шага сдвига ядра (D_{stride}, H_{stride}, W_{stride}) следующим образом:

$$D_{out} = \left[\frac{D_{in} - D_{kernel}}{D_{stride}} + 1 \right]$$
$$H_{out} = \left[\frac{H_{in} - H_{kernel}}{H_{stride}} + 1 \right] \quad (2)$$
$$W_{out} = \left[\frac{W_{in} - W_{kernel}}{W_{stride}} + 1 \right]$$

2.2.2. Слой подвыборки (Pooling layer)

Слой подвыборки или пулинга применяется для уменьшения карт признаков с сохранением наиболее важной информации. В данной модели используется слой 3Д макспулинга (3D Max Pooling Layer), его математическая формула в общем виде выглядит следующим образом. Для входной структуры данных X размерности (C, D_{in}, H_{in}, W_{in}), где C – количество каналов, D_{in} – глубина, H_{in} – высота, W_{in} – ширина. С размером окна (D_{pool}, H_{pool}, W_{pool}) и параметрами шага (D_{stride}, H_{stride}, W_{stride}):

$$Y_{c,i,j,k} = \max_{m=1\ldots D_{pool}} \max_{n=1\ldots H_{pool}} \max_{p=1\ldots W_{pool}} \times \\ \times X_{c,(i-1)\times D_{stride}+m,(j-1)\times H_{stride}+n,(k-1)\times W_{stride}+p} \quad (3)$$

Размеры выходной структуры данных (C, D_{out}, H_{out}, W_{out}) рассчитываются следующим образом:

$$D_{out} = \left[\frac{D_{in} - D_{pool}}{D_{stride}} + 1 \right]$$
$$H_{out} = \left[\frac{H_{in} - H_{pool}}{H_{stride}} + 1 \right] \quad (4)$$
$$W_{out} = \left[\frac{W_{in} - W_{pool}}{W_{stride}} + 1 \right]$$

Модель долгой-краткосрочной памяти (LSTM)

Нейронная сеть LSTM состоит из цепочки блоков (ячеек), в которые подаются признаки из последовательности. Особенностью ячейки LSTM от традиционных рекуррентных нейронных сетей является наличие дополнительного канала, который позволяет сохранять важную информацию в долгосрочной перспективе. Внутреннее устройство ячейки LSTM приведено на рис. 6.

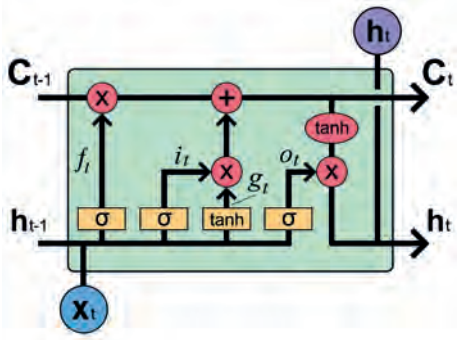


Рис. 6. Внутреннее устройство ячейки LSTM

Условные обозначения:

- C_{t-1} – вектор долгосрочного контента с предыдущего шага;
- h_{t-1} – вектор признаков скрытого состояния с предыдущего шага;
- x_t – входной вектор признаков текущего шага;
- h_t – вектор признаков скрытого состояния, полученный на текущем шаге;

C_t – вектор долгосрочного контента, полученный на текущем шаге;
σ – слой нейронной сети с сигмоидальной функцией активации;
$tanh$ – слой нейронной сети с функцией активации гиперболический тангенс;
$\oplus$ – поэлементное сложение;
$\otimes$ – поэлементное умножение;
$tanh$ – поэлементное вычисление гиперболического тангенса.

Вычисления, происходящие в ячейке LSTM, где W_j – весовые коэффициенты слоя j, [a,b] – объединение векторов a и b и b_j – это смещение (bias) на слое j, выглядят следующим образом:

1. Для фазы утраты информации (забывание ненужного):

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f$$

2. Для фазы сохранения информации (запоминание нового):

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i),$$
$$g_t = tanh(W_g * [h_{t-1}, x_t] + b_g)$$

3. Для фазы нового состояния (формирование выходного значения в канале долгосрочного контента и выходного скрытого состояния):

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o),$$
$$C_t = f_t \otimes C_{t-1} \oplus i_t \otimes g_t, \quad (5)$$
$$h_t = o_t \otimes tanh(C_t)$$

2.2.4. Полносвязный слой (Dense layer)

Стандартный полносвязный нейронный слой состоит из связанных между собой перцептронов. В общем случае, математическая формула данного слоя выглядит следующим образом:

$$Y = \sum_{i=1}^{Amount_{prev\ layer\ neurons}} (w_i * x_i) + bias \quad (6)$$

2.2.5. Слой исключения (Dropout)

Специализированный слой регуляризации нейронной сети, используемый для предотвращения или уменьшения влияния эффекта переобучения сети. Данный слой работает так, что во время процесса обучения, он случайным образом приравнивает элементы входной последовательности к нулю с вероятностью, заданной параметром, тем самым предотвращая взаимоадаптацию нейронов.

2.2.6. Функции активации

В разрабатываемой модели предполагается использование следующих функций активаций: выпрямитель (ReLU), сигмоидальная функция (Sigmoid), гиперболический тангенс tanh. Данные функции применяются к каждому значению входной структуры данных, представленной изображением или тензором, при преобразовании входных данных в выходные с помощью заданной функции.

Функция активации ReLU применяется после каждого слоя свертки, а также после каждого полносвязного слоя, кроме выходного, в сверточной нейронной сети. График функции представлен на рис. 7. Популярность ее применения в глубоких сверточных нейронных сетях обусловлена ее нетребовательностью к вычислительным ресурсам за счет выполнения простых математических операций, разреженностью активации, а также возможностью роста значения на выходе нейрона, увеличивающего влияние активации этого нейрона.

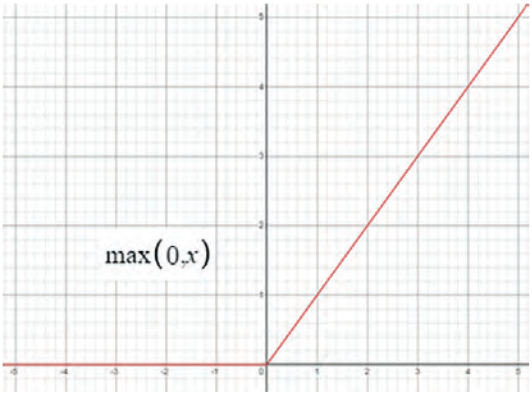


Рис. 7. График функции активации ReLU

Сигмоидальная функция активации преобразует входное значение, лежащие в интервале от отрицательной бесконечности до положительной бесконечности в значение от 0 до 1. В разрабатываемой модели данная функция используется в ячейке LSTM. Данная функция за счет своей области значений в блоке LSTM выступает в качестве «ключа», регулируя, какие данные будут интегрированы в канал хранения долгосрочного контента. График функции представлен на рис. 8.

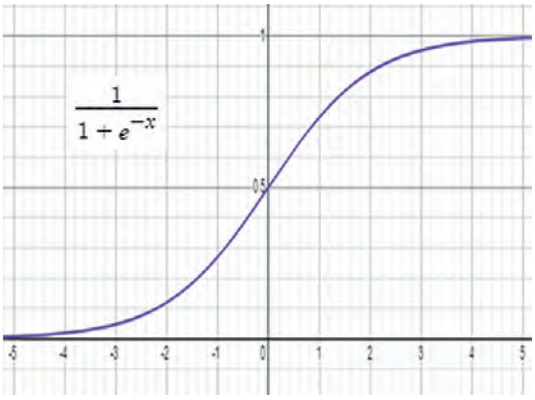


Рис. 8. График функции активации Sigmoid

Функция активации гиперболический тангенс также применяется в ячейке LSTM. Ее применение в паре с сигмоидальной функцией обусловлено тем, что на практике сходимость решения происходит быстрее и уменьшается влияние проблемы затухания градиента. Ее выходные значения лежат в диапазоне от -1 до 1, а график функции представлен на рис. 9.

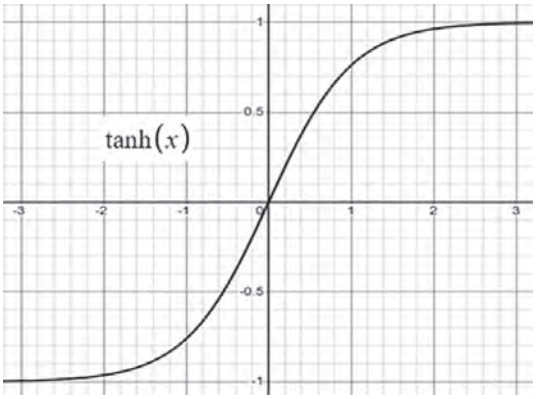


Рис. 9. График функции активации tanh

В данной работе решается задача много классовой классификации, в контексте которой широко применяется функция активации Softmax. Данная функция часто используется в выходном нейронном слое для нормализации значений слоя в вероятностное распределение, где нейрон с наибольшим значением будет иметь высокое значение вероятности. Формула расчета значений по входному вектору размерности n, для каждого значения x_i приведена ниже:

$$softmax(x_i) = \frac{exp(x_i)}{\sum_{j=1}^{n} exp(x_j)} \quad (7)$$

На этапе написания программного кода и реализации предложенной в статье модели, было принято решение об отказе использования данной функции активации из-за особенностей работы используемой функции потерь CrossEntropy из библиотеки машинного обучения PyTorch. В ходе тестирования выявилось, что при использовании данной функции скорость обучения снижается, а также значение потери на эпохах обучения не стремится к нулю, а имеет более высокий порог остановки. Также, в различных примерах, приведенных на официальных ресурсах[8], в которых используется данная библиотека, видно, что исследователи не всегда применяют ее в своих моделях.

3. Программная реализация и экспериментальные результаты

Программная реализация описанной ранее модели состоит из следующих этапов:

- импорт необходимых библиотек;
- реализация необходимых классов и функций для работы с набором данных;

8 1. Training a Classifier // PyTorch.org : сайт. – URL: https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html (дата обращения: 20.12.2023).
2. What is torch.nn really? // PyTorch.org : сайт. – URL: https://pytorch.org/tutorials/beginner/nn_tutorial.html (дата обращения: 21.12.2023).

```
class DFEW_DataSet(Dataset):
      def __init__(self,root_path_of_part_directory,path_of_excel_annotation,transform=None):
            self.root_path = root_path_of_part_directory
            self.path_of_excel_annotation=path_of_excel_annotation
            xls = pd.ExcelFile(path_of_excel_annotation)
            sheetX = xls.parse(0)
            self.length=len(sheetX["video_name,label"])
            self.transform = transform
            anno_list=[]
            for i in range(len(sheetX["video_name,label"])):
                  anno_list.append(str(sheetX["video_name,label"][i]).split("."))
            self.anno_list=anno_list.copy()
      def __len__(self):
            return self.length
      def __getitem__(self, idx):
            name,label = self.anno_list[idx]
            while(len(name)!=5):
                  name="0"+name
            path_to_item=self.root_path+"\\"+name
            res = frames_from_directory_reading(path_to_item)
            for i in range(len(res)):
                  res[i] = self.transform(res[i])
            res = torch.stack(res, 0)
            sample = (res,torch.as_tensor(dfew_classes_labels_union[int(label)-1]))
            return sample
```

Схема 1

– трансформация данных в требуемый вид для обучения и тестирования;
– построение модели нейронной сети, обучение, тестирование и вывод оценочных показателей.

При написании программной части работы были задействованы следующие библиотеки: PyTorch (библиотека, предоставляющая возможности осуществления машинного обучения), OpenCV (библиотека, предоставляющая возможности работы с фото, видеофайлами и другой графической информацией), numpy (библиотека, предоставляющая возможности для математических вычислений и преобразований, а также работы с многомерными массивами данных), matplotlib (библиотека, предоставляющая возможности для построения графиков), pandas (библиотека, предоставляющая возможности для работы с таблицами, в том числе и с excel-таблицами), tqdm (библиотека, предоставляющая возможности для визуализации прогресса работы при итерации по массивам данных).

Для осуществления возможности использования стороннего (не встроенного в саму библиотеку, как класс) набора данных, необходимо реализовать собственный класс, наследуемый от класса torch.utils.data. Dataset, в котором будут переопределены методы __len__ и __getitem__ (Схема 1).

Для перевода данных в вид, подходящий для загрузки данных на вход нейронной сети для процесса обучения и тестирования, необходимо определить данные, представленные объектом класса Dataset, как объект класса torch.utils.data.DataLoader. Данное преобразование позволяет формировать батчи данных, а также итерироваться по набору данных, работая только с определенным количеством объектов, загруженных в оперативную память. Далее приведем фрагмент примера кода, выполняющего данные преобразования для тестовой части набора данных:

```
test_data = DFEW_DataSet(…params for class…)
batch_size_var=8
test_loader = DataLoader(test_data, batch_
size=batch_size_var, shuffle = False)
```

После работы с данными, наступает этап построения нейросетевой модели. Библиотека PyTorch позволяет строить нейронные сети различными способами, предлагая разработчикам-исследователям разный уровень контроля над такими параметрами сети, как количество слоев, размерности данных на входе и выходе каждого слоя, функции активации слоев и т.д. В текущем варианте исполнения был выбран способ, в котором реализуется класс нейронной сети с двумя методами: __init__ и forward. В методе __init__ определяется каждый слой сети, со своими требуемыми параметрами, в методе forward расписывается порядок прохождения данных, а также применяемые функции активации к значениям, полученным с выходов слоев. Сокращенный пример фрагмента кода демонстрирует данный способ (Схема 2).

Далее идет реализация цикла обучения. Для тренировки нейронной сети использовался стандартный цикл, взятый из примеров документации к библиотеке. Перед циклом определяются: количество эпох, алгоритм оптимизации и функция потерь. Затем начинается цикл обучения, фрагмент которого приведен далее:

```
for epoch in tqdm(range(num_epochs)):
   net.train()
   running_loss = 0.0
   for i, batch in enumerate(tqdm(train_loader)):
      x_batch, y_batch = batch
      x_batch=x_batch.to(device)
      y_batch=y_batch.to(device)
```

```
class CNN3D_LSTM(nn.Module):
      def __init__(self):
            super().__init__()
                  # 1
                  self.conv1_1 = nn.Conv3d(3,
                  64, kernel_size=(1,3,3), padding="same")
                  self.pool1 = nn.MaxPool3d((1,2,2),(1,2,2))
                  self.dropout1 = nn.Dropout(p=0.25, inplace=False)
                  # 2
                  …

      def forward(self, x):
            x = x.transpose(1, 2)
            x = F.relu(self.conv1_1(x))
            x = self.pool1(x)
            x = self.dropout1(x)
            …
            return x
```

Схема 2

```
optimizer.zero_grad()
y_pred = net(x_batch)
loss=loss_fn(y_pred,y_batch)
loss.backward()
optimizer.step()
```

Отметим основные функции, участвующие в цикле. Функция net.train() – переводит нейронную сеть в режим обучения. В режиме обучения у сети вычисляются градиенты и работают вспомогательные слои, предотвращающие эффект переобучения, наподобие nn.Dropout(). Функция optimizer.zero_grad() – вызывается для обнуления рассчитанных на предыдущем шаге градиентов, чтобы не было эффекта «накопления». Функция loss.backward() – вызывается для вычисления градиентов. Функция optimizer.step() – вызывается для осуществления обновления весовых коэффициентов сети.

После цикла обучения идет цикл тестирования нейронной сети и вычисления различных оценочных метрик. Фрагмент основного цикла тестирования для вычисления точности работы сети приведен далее:

```
with torch.no_grad():
   net.eval()
   for data in tqdm(test_loader):
      x_batch, y_batch = data
      x_batch = x_batch.to(device)
      y_batch = y_batch.to(device)
      y_pred = net(x_batch)
      _, predicted = torch.max(y_pred.data, 1)
      total += y_batch.size(0)
      correct += (predicted == y_batch).sum().
item()
```

Основным отличием от цикла обучения является применение здесь следующих функций: torch.no_grad() – вызывается в контекстном менеджере для отключения вычисления градиентов, так как в этом нет необходимости на этапе тестирования, что экономит вычислительные и временные ресурсы; net.eval() – переводит нейронную сеть в режим оценки, отключая работу вспомогательных слоев, предотвращающих эффект переобучения, наподобие nn.Dropout().

Процесс обучения модели производился с применением параметров, указанных в таблице 2.

Обучение производилось с переносом вычислительной логики на GPU при помощи технологии CUDA на видеокарте NVIDIA GeForce RTX 3080. График потерь на каждой эпохе представлен на рис. 10.

Показатели точности распознавания вычислялись по всей тестовой выборке, а также отдельно по классам в ней. Итоговая общая точность алгоритма, подсчитанная с использованием метрики accuracy, составила: 87%. Показатели точности распознавания целевых классов представлены на рис. 11.

Для оценки эффективности решения поставленной задачи, предлагаемой нейросетевой модели,

Таблица 2

Параметры для обучения нейронной сети

№	Параметр	Значение
1	Размер батча	8
2	Количество обрабатываемых кадров, взятых из видео	20
3	Алгоритм оптимизации	Adam
4	Функция потерь	Cross Entropy
5	Количество эпох	50
6	Параметр p для слоя исключения Dropout внутри сверточных слоев	0.25
7	Параметр p для слоя исключения Dropout внутри полносвязных слоев	0.5
8	Параметр скорости обучения	0.0001

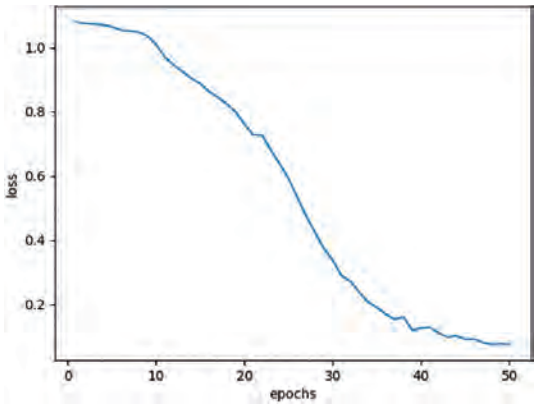


Рис. 10. График потерь на каждой эпохе

```
Accuracy for class: Normal is 90.3 %
Accuracy for class: Weak Stress is 83.1 %
Accuracy for class: Strong Stress is 87.8 %
```

Рис. 11. Показатели точности распознавания целевых классов

были также реализованы структуры с применением базовых сетей 3DCNN и LSTM. На каждой из структур был проведен процесс обучения, а затем тестирования, с вычислением общей точности работы алгоритма распознавания. В ходе адаптации базовых моделей к решаемой задаче были осуществлены следующие действия: для 3DCNN архитектуры после сверточных слоев и перед выпрямлением карт активации для подачи на вход полносвязного слоя применялся метод усреднения 3D тензора по оси количества кадров. Для LSTM архитектуры была уменьшена размерность кадров со 128 пикселей до 45, а также уменьшено количество каналов с 3 до 1, с тем чтобы количество параметров сети позволяло производить обучение за целесообразное время и объем памяти, занимаемый ими, был меньше

Таблица 3

Решение поставленной задачи с помощью разных структур нейронных сетей

Метод	Кол-во эпох	Размер батча	Функция потерь	Алгоритм оптимизации	Количество обучаемых параметров	Точность
3DCNN	50	8	Cross Entropy	Adam	5,027,715	48%
LSTM	50	8	Cross Entropy	Adam	13,549,571	45%
3DCNN+LSTM	50	8	Cross Entropy	Adam	7,915,395	87%

оперативной памяти компьютера. Полученные результаты точности работы распознавания с применением разных архитектур нейронных сетей приведены в таблице 3.

Также, для визуализации точности работы предлагаемой модели была построена матрица путаницы (confusion matrix). Каждая строка матрицы отражает целевой класс эмоций, а каждый столбец – прогнозируемый моделью класс эмоций. Матрица путаницы представлена на рис. 12.

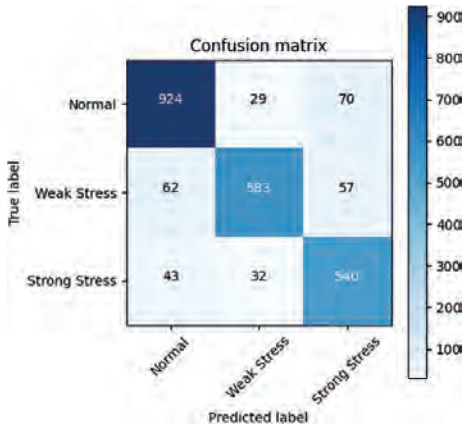


Рис. 12. Матрица путаницы

Экспериментальный результат работы построенной модели показал, что представленный метод позволяет решить задачу обнаружения негативного (сильного стрессового) состояния человека по видеоданным его лица. Точность работы алгоритма является удовлетворительной для эксперимента, так как для обучения использовался набор данных, составленный из «in the wild» видеоклипов, в которых присутствуют различные повороты головы, артефакты в виде затемнения или световых бликов на лицевой области и т.п.

Структура данной модели подходит для работы с графической информацией, представленной последовательностью изображений. Скорость обучения вне учета параметра learning rate прямо пропорционально зависит от размерности признаков на входном и скрытом слоях ячеек LSTM, так как при текущих параметрах сети данный блок содержит наибольшее количество обучаемых параметров по сравнению с другими слоями. Использование в полносвязных слоях блоков исключения (dropout) позволило избежать переобучения сети.

Заключение

В данной статье представлен метод выявления транзакций, совершаемых клиентами, подвергнутыми психологическому воздействию мошенников, на основе использования нейросетевой модели распознавания эмоций. Архитектура модели составлена из комбинации сверточной нейронной сети (3DCNN) и рекуррентной нейронной сети (в качестве долгой краткосрочной памяти (LSTM)). Данная модель позволяет работать с последовательностью изображений лица, полученных из видеоданных. Построение нейронной сети, ее обучение и тестирование были реализованы на языке Python с использованием библиотеки PyTorch. Экспериментальные результаты подтвердили работоспособность данного подхода и показали перспективность его применения для определения негативного (сильного стрессового) эмоционального состояния человека с целью отнесения совершенных им транзакций к категории подозрительных, поскольку данное состояние может быть обусловлено психологическим воздействием мошенников. Данный метод наряду с методами распознавания мошеннического воздействия по телефону, а также антифрод-системами, позволит выстроить много эшелонированную защиту пользователей банковских услуг, способную выявлять подозрительные транзакции и рассматривать их в особом порядке.

Дальнейшими исследованиями данной тематики является уменьшение входной размерности признаков в блоке LSTM с сохранением точности работы алгоритма для увеличения скорости обучения и уменьшения занимаемого объема памяти весами нейронной сети за счет уменьшения обучаемых параметров. Планируется повышение точности за счет использования других наборов данных, таких же, как и DFEW, составленных «in the wild». Далее, планируется анализ и изучение возможности замены блока LSTM на bi-LSTM или GRU, а также расширение архитектуры сети для обработки входных данных, представленных координатами ключевых точек лица.

Литература

1. *Ahmad F. M. Mansor, Ahmad A. Zainuddin, Zulkeflee Khalidin. Patient Monitoring System using Computer Vision for Emotional Recognition and Vital Signs Detection. // ResearchGate: портал. – 2020. – URL: https://www.researchgate.net/publication/344399775_Patient_Monitoring_System_using_Computer_Vision_for_Emotional_Recognition_and_Vital_Signs_Detection (дата обращения: 02.12.2023).*
2. *Mei Wang, Ziyang Huang, Yuancheng Li, Lihong Dong, Hongguang Pan. Maximum weight multi-modal information fusion algorithm of electroencephalographs and face images for emotion recognition // Computers & Electrical Engineering. – 2021. – Vol. 94. – DOI: 10.1016/j.compeleceng.2021.107319. – ISSN 0045-7906.*
3. *Cuiting Xu, Chunchuan Yan, Mingzhe Jiang, Fayadh Alenezi, Adi Alhudhaif, Norah Alnaim, Kemal Polat, Wanqing Wu. A novel facial emotion recognition method for stress inference of facial nerve paralysis patients // Expert Systems with Applications. – 2022. – Vol. 197. – DOI: 10.1016/j.eswa.2022.116705. – ISSN 0957-4174.*
4. *Shichuan Du, Aleix M. Martinez. Compound facial expressions of emotion: from basic research to clinical applications // Dialogues in Clinical Neuroscience. – 2015. – № 17:4. – Pages 443–455. – DOI: 10.31887/DCNS.2015.17.4/sdu.*
5. *Zhongshan Chen, Xinning Feng, Shengwei Zhang. Emotion detection and face recogni-tion of drivers in autonomous vehicles in IoT platform // Image and Vision Computing. – 2022. – Vol. 128. – DOI: 10.1016/j.imavis.2022.104569. – ISSN 0262-8856.*
6. *Zepf Sebastian, Hernandez Javier, Schmitt Alexander, Minker Wolfgang, Picard Rosalind. Driver Emotion Recognition for Intelligent Vehicles: A Survey // ACM Com-puting Surveys. – 2020. – DOI: 10.1145/3388790.*
7. *Lian H, Lu C, Li S, Zhao Y, Tang C, Zong Y. A Survey of Deep Learning-Based Mul-timodal Emotion Recognition: Speech, Text, and Face // Entropy (Basel). – 2023. – № 25(10):1440. – DOI: 10.3390/e25101440.*
8. *Prameela Naga, Swamy Das Marri, Raiza Borreo. Facial emotion recognition methods, datasets and technologies: A literature survey // Materials Today: Proceedings. – 2023. – Vol. 80. – Pages 2824–2828. – DOI: 10.1016/j.matpr.2021.07.046.*
9. *Chahak Gautam, K. R Seeja. Facial emotion recognition using Handcrafted features and CNN // Procedia Computer Science. – 2023. – Vol. 218. – Pages 1295–1303. – DOI: 10.1016/j.procs.2023.01.108.*
10. *Zia Ullah, Lin Qi, Asif Hasan, Muhammad Asim. Improved Deep CNN-based Two Stream Super Resolution and Hybrid Deep Model-based Facial Emotion Recognition // Engineering Applications of Artificial Intelligence. – 2022. – Vol. 116. – DOI: 10.1016/j.engappai.2022.105486.*
11. *Elham S. Salama, Reda A. El-Khoribi, Mahmoud E. Shoman, Mohamed A. Wahby Shalaby. A 3D-convolutional neural network framework with ensemble learning tech-niques for multi-modal emotion recognition // Egyptian Informatics Journal. – 2021. – Vol. 22. – Issue 2. – Pages 167–176. – DOI: 10.1016/j.eij.2020.07.005.*
12. *Radha Priyadharsini G, Krishnaveni K. A novel framework using binary attention mechanism based deep convolution neural network for face emotion recognition // Measurement: Sensors. – 2023. – Vol. 30. – DOI: 10.1016/j.measen.2023.100881.*
13. *Anjali R, J. Babitha, Rithika W, Ms.Reeja S.L. Stress Detection Based on Emotion Recognition Using Deep Learning // National Conference on Smart Systems and Tech-nologies. – 2021. – Vol. 8. – Issue 7. – Pages 109–114.*
14. *Orhan Atila, Abdulkadir Şengür. Attention guided 3D CNN-LSTM model for accu-rate speech based emotion recognition // Applied Acoustics. – 2021. – Vol. 182. – DOI: 10.1016/j.apacoust.2021.108260.*
15. *Wafa Mellouk, Wahida Handouzi. CNN-LSTM for automatic emotion recognition using contactless photoplythesmographic signals // Biomedical Signal Processing and Control. – 2023. – Vol. 85. – DOI: 10.1016/j.bspc.2023.104907.*
16. *El Mehdi Saoudi, Jaafar Jaafari, Said Jai Andaloussi. Advancing human action recog-nition: A hybrid approach using attention-based LSTM and 3D CNN // Scientific Afri-can. – 2023. – Vol. 21. – DOI: 10.1016/j.sciaf.2023.e01796.*
17. *Emanuel Di Nardo, Vincenzo Santopietro, Alfredo Petrosino. Emotion recognition at the edge with AI specific low power architectures // Microprocessors and Microsys-tems. – 2021. – Vol. 85. – DOI: 10.1016/j.micpro.2021.104299.*
18. *Yi Chen, Jun Bin, Chao Kang. Application of machine vision and convolutional neural networks in discriminating tobacco leaf maturity on mobile devices // Smart Agricultur-al Technology. – 2023. – Vol. 5. – DOI: 10.1016/j.atech.2023.100322.*
19. *Deyuan Qu, Sudip Dhakal, Dominic Carrillo. Facial Emotion Recognition using CNN in PyTorch. – URL: https://arxiv.org/pdf/2312.10818.pdf (дата обращения: 14.12.2023).*
20. *Glen Berman. Machine Learning practices and infrastructures. – URL: https://arxiv.org/pdf/2307.06518.pdf (дата обращения: 14.12.2023).*
21. *Ramesh Naidu P, Pruthvi Sagar S, Praveen K, Kiran K, Khalandar K. Stress Recogni-tion Using Facial Landmarks and Cnn (Alexnet) // Journal of Physics: Conference Series. – 2021. – 2089(1):012039 – DOI: 10.1088/1742-6596/2089/1/012039.*
22. *Xingxun Jiang, Yuan Zong, Wenming Zheng, Chuangao Tang, Wanchuang Xia, Cheng Lu, Jiateng Liu. DFEW: A Large-Scale Database for Recognizing Dynamic Fa-cial Expressions in the Wild // Proceedings of the 28th ACM International Conference on Multimedia. – 2020. – Pages 2881–2889. – DOI: 10.48550/arXiv.2008.05924.*
23. *Andrea Scarantino. Core Affect and Natural Affective Kinds // Philosophy of Science. – 2009. – Vol. 76. – Issue 5. – Pages 940–957. – DOI: 10.1086/605816*

ОСОБЕННОСТИ РЕАЛИЗАЦИИ СИСТЕМ КРИПТОАНАЛИЗА ГОМОМОРФНЫХ ШИФРОВ, ОСНОВАННЫХ НА ЗАДАЧЕ ФАКТОРИЗАЦИИ ЧИСЕЛ, НА ПРИМЕРЕ КРИПТОСИСТЕМЫ MORE

Бабенко Л. К.[1], Стародубцев В. С.[2]

DOI: 10.21681/2311-3456-2024-3-141-145

Цель работы: *определение общих техник, тактик и процедур для различных методов криптоанализа гомоморфных шифров, основанных на задаче факторизации чисел, и разработка независимой от применяемого метода криптоанализа архитектуры системы для упрощения этого процесса путём предоставления удобного окружения и инструментов.*

Методы исследования: *анализ возможных реализаций архитектурных особенностей при создании систем криптоанализа гомоморфных шифров, основанных на задаче факторизации чисел.*

Объект исследования: *гомоморфные шифры, основанные на задаче факторизации чисел, криптосистема MORE (Matrix Operation for Randomization or Encryption), криптоанализ гомоморфных шифров, основанных на задаче факторизации чисел, особенности архитектуры систем для проведения криптоанализа гомоморфных шифров, основанных на задаче факторизации чисел при различных типах атак.*

Результаты исследования: *разработана архитектура системы криптоанализа для оценки криптостойкости рассматриваемых шифров, основанных на задаче факторизации чисел путём проведения всестороннего анализа уязвимостей для различных атак. На примере атаки с известным открытым текстом на криптосистему MORE, основанную на задаче факторизации чисел, определены общие особенности архитектуры и особенности, свойственные конкретным шифрам, основанным на задаче факторизации чисел, и конкретным типам атак.*

Практическая значимость: *реализация системы криптоанализа на основе предложенной архитектуры позволит исследователям и криптоаналитикам более подробно изучить потенциальные уязвимости в гомоморфных криптосистемах, основанных на задаче факторизации чисел, что позволит разработать более эффективные меры по укреплению стойкости таких шифров.*

Ключевые слова: *Информационная безопасность; конфиденциальная информация; гомоморфное шифрование; криптосистема MORE; криптоанализ; архитектура системы криптоанализа.*

FEATURES OF THE IMPLEMENTATION OF THE CRYPTANALYSIS SYSTEMS OF HOMOMORPHIC CIPHERS BASED ON THE PROBLEM OF FACTORIZATION OF NUMBERS, USING THE EXAMPLE OF THE CRYPTOSYSTEM MORE

Babenko L. K.[3], Starodubcev V. S.[4]

Purpose of the work: *definition of common techniques, tactics and procedures for various methods of cryptanalysis of homomorphic ciphers based on the problem of factorization of numbers, and development of a system architecture independent of the applied cryptanalysis method to simplify this process by providing a convenient environment and tools.*

1 Бабенко Людмила Климентьевна, доктор технических наук, профессор, Южный Федеральный Университет «ЮФУ», Институт компьютерных технологий и информационной безопасности, г. Таганрог, Россия. E-mail: lkbabenko@sfedu.ru

2 Стародубцев Виталий Сергеевич, студент, Южный Федеральный Университет «ЮФУ», Институт компьютерных технологий и информационной безопасности, г. Таганрог, Россия. E-mail: vstarodubcev@sfedu.ru

3 Liudmila Babenko, Dr.Sc., Professor, Southern Federal University «SFedU», Institute of Computer Technologies and Information Security, Taganrog, Russia. E-mail: lkbabenko@sfedu.ru

4 Vitalij Starodubcev, student, Southern Federal University «SFedU», Institute of Computer Technologies and Information Security, Taganrog, Russia. E-mail: vstarodubcev@sfedu.ru

Research methods: *analysis of possible implementations of architectural features in the creation of cryptanalysis systems for homomorphic ciphers based on the problem of number factorization.*

The object of research: *homomorphic ciphers based on the problem of number factorization, cryptosystem MORE (Matrix Operation for Randomization or Encryption), cryptanalysis of homomorphic ciphers based on the problem of number factorization, features of the architecture of systems for cryptanalysis of homomorphic ciphers based on the problem of number factorization in various types of attacks.*

Research results: *the architecture of a cryptanalysis system has been developed to assess the cryptographic strength of the ciphers in question, based on the task of factorizing numbers by conducting a comprehensive vulnerability analysis for various attacks. Using the example of an attack with a known plaintext on the MORE cryptosystem based on the number factorization problem, general architectural features and features peculiar to specific ciphers based on the number factorization problem and specific types of attacks are determined.*

Practical significance: *the implementation of a cryptanalysis system based on the proposed architecture will allow researchers and cryptanalysts to study in more detail potential vulnerabilities in homomorphic cryptosystems based on the problem of number factorization, which will allow developing more effective measures to strengthen the durability of such ciphers.*

Keywords: *Information security; confidential information; homomorphic encryption; cryptosystem MORE; cryptanalysis; architecture of the cryptanalysis system.*

Введение

Гомоморфное шифрование представляет собой метод защиты данных, позволяющий выполнять операции над зашифрованными данными и получать корректный результат, соответствующий операциям, выполненным над открытым текстом [1–3]. В данной статье рассматриваются гомоморфные криптосистемы, основанные на задаче факторизации чисел (Доминго-Феррера[5], Жирова А. О., Жировой О. В., Кренделева С. Ф.[6], MORE[7]), упоминаемые в [4–6]. Особенностью гомоморфных шифров, основанных на задаче факторизации чисел является то, что при их использовании для отражения различных типов атак необходимо иметь информацию о стойкости рассматриваемых шифров, то есть проводить трудоемкий криптоанализ. Разработка разнообразных средств, позволяющих облегчить проведение криптоанализа, является важной задачей. В данной работе рассматриваются актуальная задача разработки системных средств и определение архитектурных особенностей, повышающих эффективность и создающих удобства при оценке стойкости шифров для разных типов атак. Изложение основывается на рассмотрении одной из гомоморфных криптосистем, основанных на задаче факторизации чисел – криптосистеме MORE.

Рассматриваются архитектурные особенности системы криптоанализа, которая обладает универсальностью, предоставляет исследователям необходимое окружение и удобные инструменты и, как следствие, позволит сконцентрироваться на реализации собственных методов криптоанализа, а не тратить время на создание среды для проведения необходимых операций и оценки полученных результатов.

5 Domingo-Ferrer J. A provably secure additive and multiplicative privacy homomorphism //International Conference on Information Security. – Berlin, Heidelberg : Springer Berlin Heidelberg, 2002. – С. 471–483.

6 Жиров А. О., Жирова О. В., Кренделев С. Ф. Безопасные облачные вычисления с помощью гомоморфной криптографии //Безопасность информационных технологий. – 2013. – Т. 20. – №. 1. – С. 6–12.

7 Kipnis A., Hibshoosh E. Efficient methods for practical fully homomorphic symmetric-key encrypton, randomization and verification //Cryptology ePrint Archive. – 2012.

Описание криптосистемы MORE

Данная криптосистема использует шифрование открытых текстов путем их сочетания со случайной обратимой матрицей по модулю RSA в качестве ключа шифрования [7]. Благодаря применению случайного элемента при шифровании, один и тот же открытый текст может иметь разные шифртексты при использовании того же ключа [8].

Для реализации криптосистемы MORE определяются значения p и q, из которых затем формируется труднофакторизуемое число $n = p \cdot q$.

Для формирования секретного ключа необходимо создать обратимую матрицу K размером 2x2 со случайными элементами $k \in Z_n$ по формуле (1).

$$K = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix}. \quad (1)$$

Для шифрования открытого текста m необходимо сформировать случайное большое число $s \in Z_n$ по формуле (2).

$$s \xleftarrow{\$} Z_n, \quad (2)$$

где $\xleftarrow{\$}$ является операцией выбора случайного элемента.

Затем открытый текст m и сформированное случайное число s размещаются на главной диагонали матрицы A по формуле (3).

$$A = \begin{pmatrix} s & 0 \\ 0 & m \end{pmatrix}. \quad (3)$$

Шифртекст вычисляется путём умножения матрицы секретного ключа K на закодированный открытый

текст A с последующим умножением полученного результата на обратную матрицу ключа K^{-1} по формуле (4).

$$C = (K \cdot A) \cdot K^{-1}. \quad (4)$$

Важно отметить, что операция умножения матриц в общем случае некоммутативна, поэтому в формуле (4) крайне важно соблюдение порядка действий [9].

Криптосистема MORE определяет следующий перечень гомоморфных операций [6]:

1. Сложение;
2. Умножение;
3. Деление.

Для выполнения сложения достаточно сложить матрицы соответствующих шифртекстов по формуле (5).

$$C_3 = C_1 + C_2, \quad (5)$$

где C_3 – результирующий шифртекст, C_1 – первый шифртекст, C_2 – второй шифртекст.

Для выполнения умножения необходимо умножить матрицы соответствующих шифртекстов по формуле (6).

$$C_3 = C_1 \cdot C_2, \quad (6)$$

где C_3 – результирующий шифртекст, C_1 – первый шифртекст, C_2 – второй шифртекст.

Для выполнения деления необходимо умножить матрицу шифртекста C_1 на обратную матрицу шифртекста C_2 по формуле (7). Деление выполняется только при условии, что определитель матрицы C_2 не равен 0.

$$C_3 = C_1 \cdot C_2^{-1}, \quad (7)$$

где C_3 – результирующий шифртекст, C_1 – первый шифртекст, C_2 – второй шифртекст.

Для расшифрования шифртекста C необходимо умножить обратную матрицу ключа K^{-1} на матрицу шифртекста C, с последующим умножением полученного результата на матрицу секретного ключа K по формуле (8).

$$M = (K^{-1} \cdot C) \cdot K. \quad (8)$$

Поскольку полученный после умножений результат – матрица M размерами 2x2, для получения открытого текста её необходимо раскодировать – извлечь элемент m_{22} по формуле (9).

$$M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}. \quad (9)$$

Атака с известным открытым текстом на криптосистему MORE

Атака данного типа подразумевает, что криптоаналитик обладает некоторым набором пар (открытый текст, шифртекст), изготовленных на одном ключе [10].

Обозначим K матрицу ключа. Зная, как происходит генерация ключа, можно определить, что элементы главной диагонали ненулевые [11], поэтому справедливо выражение, приведенное в формуле (10).

$$K = D \cdot \begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix}, \quad (10)$$

где D – диагонально обратная матрица.

Поскольку операция умножения для диагональных матриц обладает свойством коммутативности [12], можно составить уравнение, показанное в формуле (11).

$$E_K(m) = K^{-1} \cdot \begin{pmatrix} s & 0 \\ 0 & m \end{pmatrix} \cdot K = \begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix}^{-1} \cdot D^{-1} \cdot \begin{pmatrix} s & 0 \\ 0 & m \end{pmatrix} \cdot \mathrm{D} \cdot = = \begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} s & 0 \\ 0 & m \end{pmatrix} \cdot \begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix}, \quad (11)$$

где $E_K(m)$ – операция шифрования открытого текста m на ключе K, s – случайное большое число $s \in Z_n$, сформированное на этапе шифрования.

По условию атаки с известным открытым текстом [13] криптоаналитику известно значение матрицы шифртекста, как показано в формуле (12).

$$E_K(m) = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}. \quad (12)$$

Тогда формула (12) может быть представлена в виде, приведенном в формуле (13).

$$\begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix} \cdot \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} = \begin{pmatrix} s & 0 \\ 0 & m \end{pmatrix} \cdot \begin{pmatrix} 1 & b \\ c & 1 \end{pmatrix}; \quad \begin{pmatrix} \alpha+b\gamma & \beta+b\delta \\ c\alpha+\gamma & c\beta+\delta \end{pmatrix} = \begin{pmatrix} s & sb \\ mc & m \end{pmatrix}. \quad (13)$$

Из показанного в формуле (13) уравнения можно выразить значение открытого текста m по формуле (14).

$$m = c\beta + \delta. \quad (14)$$

В формуле (14) для криптоаналитика является неизвестным только значение c, зависящее от ключа шифрования K, которое можно вычислить по формуле (15).

$$c = \frac{m - \delta}{\beta}. \quad (15)$$

Таким образом, для взлома шифра MORE при помощи атаки с известным открытым текстом, криптоаналитику достаточно обладать одной парой (открытый текст – шифртекст).

Описание особенностей архитектуры системы криптоанализа гомоморфных шифров, основанных на задаче факторизации чисел

Исходя из приведенных в формулах (10)–(15) этапов атаки с известным открытым текстом на криптосистему MORE, а также сравнения с другими атаками на шифры, основанные на задаче факторизации чисел, видно, что этапы атаки являются специфичными для конкретной криптосистемы и не могут быть объединены в единый программный модуль. Поэтому система криптоанализа предоставляет лишь интерфейсы, в рамках которых прикладной программист реализует методы криптоанализа.

Однако криптоанализ не ограничивается реализацией конкретных методов. Важными задачами

является выбор, подготовка и хранение исходных данных, отслеживание промежуточных этапов атаки, регистрация временных характеристик, отображение и сохранение результатов. Задачи подобного рода актуальны для различных атак и могут быть автоматизированы путём создания общих программных модулей, объединенных в разработанную систему криптоанализа.

Реализация системы криптоанализа гомоморфных шифров, основанных на задаче факторизации чисел выполнена на языке программирования C#10 (.NET 6.0), поскольку данный язык широко распространен, обеспечивает высокую производительность и эффективное управление ресурсами, интегрируется с другими технологиями Microsoft, такими как ASP.NET, WPF, Xamarin и т. д., что обеспечивает возможность разработки приложений для различных платформ [14].

Архитектура системы криптоанализа состоит из ядра с подключаемыми модулями. Имеется графический пользовательский интерфейс.

Ядро системы – не имеет зависимостей ни от платформы, ни от остальных модулей системы. Определяет программные модули и интерфейсы, общие для различных атак и графического пользовательского интерфейса.

Подключаемые модули реализаций конкретных атак – зависят только от ядра системы. Определяют методы реализации конкретных атак на криптосистемы, основанные на задаче факторизации чисел. Данные модули собираются в файлы DLL (Dynamic Link Library), которые затем подключаются к ядру приложения динамически с помощью контроллера атак [15].

Графический пользовательский интерфейс – зависит от платформы и ядра системы. Отвечает за отображение данных и удобное взаимодействие пользователя с ними.

Ядро системы криптоанализа определяет следующие модули:

1. Контроллер атак – определяет механизм подключения реализаций конкретных атак, собранных в файлы DLL.
2. Контроллер пакетов запусков – управляет созданием, открытием, сохранением, связыванием с реализациями атак и закрытием пакетов запусков. Пакетом запусков именуется набор запусков, связанных с подключенной атакой. Запуск атаки – это исходные данные, с которыми была проведена атака, а также результаты этой атаки (если атака выполнялась).
3. Система регистрации и учёта действий пользователя – контролирует изменения текущего пакета запусков. Любое действие с пакетом представляется объектом типа «команда», определяющим два действия: выполнить и отменить (operation-oriented механизм Undo / Redo) [16]. Первое действие выполняет изменения пакета, запрошенное пользователем, второе – откатывает пакет к предыдущему состоянию. Таким образом, любые изменения пакета являются обратимыми.
4. Логгер – записывает историю всех действий пользователя, а также событий системы в текстовый файл лога с разделением по уровням [17]. Позволяет упростить поддержку системы путем подробной записи информации о возникших исключениях, сообщениях об ошибках с трассировкой стека и предысторией действий пользователя, которая затем может быть использована для определения причины ошибки и её повторного воспроизведения [18].

Графический интерфейс системы реализован с помощью шаблона графического пользовательского интерфейса MVP (Model-View-Presenter) [19]. Модель (model) и представитель (presenter) реализованы в ядре, отображение (view) – в отдельном проекте Windows Forms [20]. Такой подход позволяет перенести систему на другую платформу изменением только одного модуля отображения (view) без необходимости внесения изменений в других модулях.

Пример использования системы для реализации атаки с известным открытым текстом на криптосистему MORE

Продемонстрируем на примере атаки с известным открытым текстом на криптосистему MORE как выглядит реализация криптоанализа с применением предложенных средств.

Пользователь запускает графический интерфейс системы криптоанализа, создает новый пакет запусков. Созданный пакет связывает с реализацией атаки на криптосистему MORE. Далее создает новый запуск и заполняет исходные данные атаки. В качестве примера использовались следующие исходные данные: $p = 13$ и $q = 17$, количество пар (открытый текст – шифртекст) – 1. Введенные пользователем данные проходят валидацию с помощью метода, предусмотренного реализацией атаки. Если данные корректны, пользователь запускает процесс атаки на криптосистему. В системе регистрации и учёта действий пользователя записывается предыдущее состояние, формируется и передается на выполнение соответствующая команда. Графический интерфейс становится неактивным для действий пользователя за исключением компонента, отображающего промежуточные данные в ходе атаки.

В качестве ключа шифрования выбран ключ

$$K = \begin{pmatrix} 49 & 27 \\ 204 & 141 \end{pmatrix}$$

и сформирована одна пара (открытый текст – шифртекст):

$$\left(146, \begin{pmatrix} -391{,}5310 & 102{,}9315 \\ -2237{,}8844 & 574{,}5310 \end{pmatrix}\right).$$

Значение c, зависящее от ключа шифрования K вычислено по формуле (16).

$$c = \frac{m - \delta}{\beta} = \frac{146 - 574{,}5310}{102{,}9315} = -4{,}1633. \qquad (16)$$

Для проверки корректности найденного значения с сформирована другая пара (открытый текст – шифртекст) на том же ключе:

$$\left(121, \begin{pmatrix} -41{,}7388 & 31{,}1627 \\ -677{,}5246 & 250{,}7388 \end{pmatrix}\right).$$

Шифртекст из созданной пары расшифрован с помощью значения с по формуле (17).

$$m' = c\beta + \delta = (-4{,}1633) \cdot 31{,}1627 + 250{,}7388 = 121. \quad (17)$$

Найденное в результате атаки значение c позволяет получить соответствующий открытый текст $m' = m = 121$ (формула (17)), что доказывает успешность проведенной атаки.

По завершении процесса атаки в графическом пользовательском интерфейсе отображаются результаты атаки: временные характеристики проведенной атаки, количество итераций и атомарных операций, а также полная история всех шагов. Кроме того, полученные результаты автоматически сохраняются в пакет запусков для обеспечения возможности просмотра без необходимости повторного запуска атаки.

Выводы

В данной работе проведен анализ симметричной гомоморфной криптосистемы MORE, а также атаки с известным открытым текстом на этот шифр. На основе проведенного исследования на примере криптосистемы MORE определено, какие действия являются специфичными для конкретной криптосистемы, а какие могут быть автоматизированы путём создания общих программных модулей, объединенных в систему криптоанализа гомоморфных шифров, основанных на факторизации чисел.

Представлено описание основных модулей архитектуры системы криптоанализа гомоморфных шифров: ядро, графический пользовательский интерфейс и интерфейс подключаемых модулей. Для каждого модуля системы приводится описание подхода к реализации, а также преимущества применения данного подхода.

Литература

1. *Минаков С. С. Основные криптографические механизмы защиты данных, передаваемых в облачные сервисы и сети хранения данных // Вопросы кибербезопасности. – 2020. – № 3(37). – С. 66–75. DOI: 10.21681/2311-3456-2020-03-66-75*
2. *Гаража А. А., Герасимов И. Ю., Николаев М. В., Чижов И. В. Об использовании библиотек полностью гомоморфного шифрования // International Journal of Open Information Technologies. – 2021. – Т. 9, № 3. – С. 11–22.*
3. *Шачина В. А. Гомоморфная криптография в базах данных // Прикладная математика и информатика: современные исследования в области естественных и технических наук: Материалы V Международной научно-практической конференции (школы-семинара) молодых ученых, Тольятти, 22–24 апреля 2019 года. – 2019. – С. 468–473.*
4. *Hariss K., Noura H., Samhat A. E. An efficient fully homomorphic symmetric encryption algorithm // Multimedia Tools and Applications. – 2020. – Т. 79. – №. 17. – С. 12139-12164. DOI:10.1007/s11042-019-08511-2*
5. *Иванов А. И., Сулавко А. Е. Проект третьего национального стандарта России по быстрому автоматическому обучению больших сетей корреляционных нейронов на малых обучающих выборках биометрических данных // Вопросы кибербезопасности. – 2021. – № 3 (43). – С. 84–93. DOI: 10.21681/2311-3456-2021-3-84-93*
6. *Sana M. U. et al. Enhanced security in cloud computing using neural network and encryption // IEEE Access. – 2021. – Т. 9. – С. 145785–145799. DOI:10.1109/ACCESS.2021.3122938*
7. *Тришин А. Е. Атака Винера и слабые ключи криптосистемы RSA // Дискретная математика. – 2023. – Т. 35. – №. 3. – С. 71–80. DOI: 10.4213/dm1773*
8. *Трепачева А. В. О стойкости гомоморфной криптосистемы Доминго-Феррера против атаки только по шифртекстам // Прикладная дискретная математика. Приложение. – 2023. – № 16. – С. 98–102. DOI: 10.17223/2226308X/16/25*
9. *Гантмахер Ф. Теория матриц. – Litres, 2022. 576 с.*
10. *Горохов Н. Б., Преображенский Ю. П. Об особенностях криптографических систем защиты информации // Молодежь и XXI век-2022. – 2022. – С. 43–46.*
11. *Vaudenay D. V. S. Cryptanalysis of enhanced more //Tatra Mt. Math. Publ. – 2019. – Т. 73. – С. 163-178. DOI: 10.2478/tmmp-2019-0012*
12. *Винберг Э. Курс алгебры. – Litres, 2022. 592 с.*
13. *Yuan Y., Mo Y. L. Security for cyber-physical systems: Secure control against known-plaintext attack // Science China Technological Sciences. – 2020. – Т. 63. – №. 9. – С. 1637–1646. DOI: 10.1007/s11431-020-1621-y*
14. *Bahar A. Y. et al. Survey on Features and Comparisons of Programming Languages (PYTHON, JAVA, AND C#) // 2022 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS). – IEEE, 2022. – С. 154-163. DOI: 10.1109/ICETSIS55481.2022.9888839*
15. *Нагибин В. А. Проектирование и реализация системы подключаемых модулей в приложениях на языке C# // Путь в науку: прикладная математика, информатика и информационные технологии. – 2023. – С. 27–29.*
16. *Jeong J., Zeng J., Jung C. Capri: Compiler and architecture support for whole-system persistence // Proceedings of the 31st International Symposium on High-Performance Parallel and Distributed Computing. – 2022. – С. 71-83. DOI: 10.1145/3502181.3531474*
17. *Волушкова В. Л. Многоуровневое логгирование работы процессов и задач // ИТНОУ: информационные технологии в науке, образовании и управлении. – 2021. – №. 1 (17). – С. 60–64. DOI: 10.47501/ITNOU.2021.1.060-064*
18. *Киптенко А. В., Бахарева Н. Ф. Отладка программного обеспечения с помощью лог файлов //Актуальные проблемы информатики, радиотехники и связи. – 2023. – С. 157–158.*
19. *Jánki Z. R., Bilicki V. Rule-Based Architectural Design Pattern Recognition with GPT Models // Electronics. – 2023. – Т. 12. – №. 15. – С. 3364. DOI: 10.3390/electronics12153364*
20. *Pasztaleniec M., Skublewska-Paszkowska M. Comparative analysis of Windows Presentation Foundation and Windows Forms // Journal of Computer Sciences Institute. – 2020. – Т. 14. – С. 26-30. DOI: 10.35784/jcsi.1571*

CYBERSECURITY ISSUES

SCIENTIFIC PEER-REVIEWED JOURNAL

2024, № 3 (61)

Cybersecurity Issues is a research periodical scientific and practical publication specializing in information security.
Published six times a year

https://cyberrus.info

The journal is being published from 2013
(Registration Certificate PI No. FS 77-75239).
CrossRef number (DOI): 10.21681/2311-3456

The journal is included in the Russian list of peer-reviewed academic publications of the Higher Attestation Commission (VAK), it is registered in the Russian Science Citation Index (RSCI/RINTs) on the Web of Science (WoS) platform and holds the 1st place in its cyber security rating. The journal's articles are available in full text

Founder and publisher
JSC «NPO «Echelon»
Postal address: Elektrozavodskaya str., 24, bld. 1, 107023, Moscow, Russia
E-mail: editor@cyberrus.info

CONTENTS

Эшелон
комплексная безопасность

Сканер-ВС
анализ защищенности

СКАНИРОВАНИЕ НА УЯЗВИМОСТИ НИКОГДА НЕ БЫЛО ТАКИМ БЫСТРЫМ!

ГК «Эшелон» представляет новый релиз системы управления уязвимостями Сканер-ВС 6. Сканер-ВС используется более чем в 5 000 организаций в России и позволяет как проводить периодическое сканирование на поиск уязвимостей, так и организовать непрерывный контроль защищенности.

Решение является ключевым компонентом, позволяющим внедрить эффективный процесс управления уязвимостями.

Скачать демо-версию «Сканер-ВС 6» (количество IP: 16, пробный период: 2 месяца) можно на сайте продукта: https://scaner-vs.ru/.

Получить техническую консультацию в группе продукта в телеграм: https://t.me/scanervs

Высокая скорость поиска

Сканер-ВС 6 обладает высокой скоростью поиска уязвимостей благодаря технологии «без скриптов»

Актуальная база уязвимостей

Ежедневно обновляемая база данных уязвимостей позволяет держать руку на пульсе последних изменений

Комплексный подход

Комплексное тестирование защищенности позволяет выявлять максимальное количество нарушений ИБ

Работа в защищенной среде

Работа в среде защищенной операционной системы Astra Linux 1.7

Отчетность

Единая среда для проведения тестирования и формирования отчетов, содержащих различную информацию в зависимости от степени детализации

Исполнение

Наличие исполнений в виде дистрибутива под Astra Linux 1.7 и LiveUSB с предустановленной ОС и с поддержкой режима сохранения изменений.

CYBERSECURITY ISSUES
VOPROSY KIBERBEZOPASNOSTI

№3 2024

DOI: 10.21681/2311-3456

- Conceptual Issues in Cybersecurity
- Secure Artificial Intelligence
- Information Security Audit

www.cyberrus.info
editor@cyberrus.info