МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное
учреждение высшего образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт
Компьютерных
Технологий и
Информационной
Безопасности

СОВРЕМЕННЫЕ МЕТОДЫ, СРЕДСТВА И ТЕХНОЛОГИИ ЗАЩИТЫ ИНФОРМАЦИИ – 2025

Сборник трудов

XVI Международной научно-практической
конференции имени Олега Борисовича Макаревича

Таганрог, 19–22 мая 2025 г.

Ростов-на-Дону – Таганрог
Издательство Южного федерального университета
2025

УДК 004.56(063)
ББК 16.8 я431
С56

С56 **Современные методы, средства и технологии защиты информации – 2025** [Электронный ресурс] : сборник трудов XVI Международной научно-практической конференции имени Олега Борисовича Макаревича (Таганрог, 19–22 мая 2025 г.) ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2025. – Электрон. текстовые дан. (1 файл: 2,74 Мб). – 1 электрон. опт. диск (CD-R). – Системные требования: процессор с тактовой частотой 1,5 ГГц и выше, 2 Гб оперативной памяти, Windows 7 SP1/8, 8.1/10 (32- и 64-разрядные версии), Windows 11 (64-разрядная версия), Acrobat Reader, привод DVD-ROM. – Загл. с экрана. – 117 с.
ISBN 978-5-9275-5088-3

В сборник трудов XVI Международной научно-практической конференции имени Олега Борисовича Макаревича вошли статьи по следующим направлениям: «Методы и системы информационной безопасности»; «Вопросы информационной безопасности автоматизированных систем и систем связи»; «Методы сетевой безопасности»; «Безопасность распределенных систем и телекоммуникаций»; «Безопасность критических информационных инфраструктур»; «Теоретические и практические аспекты криптографии»; «Безопасность киберфизических систем и БПЛА»; «Безопасность программного обеспечения»; «Правовые основы и защита государственной, коммерческой тайны и интеллектуальной собственности»; «Новые нормативные правовые и методические документы, регламентирующие деятельность по защите информации и объектов»; «Подготовка специалистов в области информационной безопасности»; «Безопасность систем Искусственного Интеллекта».

Материалы публикуются в авторской редакции

ISBN 978-5-9275-5088-3

УДК 004.56(063)
ББК 16.8 я431

ПРОГРАММНЫЙ КОМИТЕТ

Председатель

Веселов Г.Е. – д.т.н., доцент, директор Института компьютерных технологий и информационной безопасности (ИКТИБ) Южного федерального университета (ЮФУ), Россия.

Заместитель председателя

Бабенко Л.К. – д.т.н., профессор, профессор кафедры «Безопасность информационных технологий» (БИТ) им. О.Б. Макаревича ИКТИБ ЮФУ, Россия.

ЧЛЕНЫ ПРОГРАММНОГО КОМИТЕТА

Atilla Elci – Professor, PhD, Hasan Kalyoncu University, Türkiye;

Luis Ramiro Piñeiro Díaz – Professor, PhD, Cryptography Institute of the Faculty of Mathematics and Computing of the University of Havana, Cuba;

Maxim Anikeev – Associate Professor, PhD, Fraunhofer SIT | ATHENE, researcher, Germany;

Miguel Katrib Mora – Professor, PhD, Cryptography Institute of the Faculty of Mathematics and Computing of the University of Havana, Cuba;

Mohd Helmy Abd Wahab – Professor, PhD, Universiti Tun Hussein Onn Malaysia;

Pradeep Kumar Singh – Professor, PhD, Central University of Jammu, India;

Абрамов Е.С. – к.т.н., доцент, заведующий кафедрой БИТ им. О.Б. Макаревича ИКТИБ ЮФУ, Россия;

Алгазы К.Т. – с.н.с., PhD, Институт информационных и вычислительных технологий комитета науки министерства науки и высшего образования Республики Казахстан (ИИВТ КН МНВО РК), Казахстан;

Белов Е.Б. – председатель ФУМО ВО ИБ, Россия;

Ищукова Е.А. – к.т.н., доцент, доцент кафедры БИТ им. О.Б. Макаревича ИКТИБ ЮФУ, Россия;

Калмыков И.А. – д.т.н., профессор, профессор кафедры вычислительной математики и кибернетики Северо-Кавказского федерального университета (СКФУ), Россия;

Капалова Н.А. – г.н.с., ИИВТ КН МНВО РК, Казахстан;

Климов С.М. – д.т.н., профессор, 4-й Центральный научно-исследовательский институт, Россия;

Конявский В.А. – д.т.н., заведующий кафедрой «Информационная безопасность» Московского физико-технического Института (МФТИ), Россия;

Котенко И.В. – д.т.н., профессор, Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН), Россия;

УДК 004.089

М.А. Болатбек, Ш.Ж. Мусиралиева

Казахский национальный университет имени аль-Фараби,
Казахстан, г. Алматы

ОБНАРУЖЕНИЕ ДЕСТРУКТИВНОЙ РЕЧИ В СОЦИАЛЬНЫХ СЕТЯХ С ПОМОЩЬЮ ГРАФОВЫХ НЕЙРОННЫХ СЕТЕЙ

В цифровую эпоху распространение разжигающих ненависть высказываний стало насущной глобальной проблемой, усугубляемой быстрым распространением контента на онлайн-платформах и в социальных сетях. В этом исследовании рассматриваются передовые методики выявления деструктивной речи посредством интеграции методов обработки естественного языка (NLP) и машинного обучения. В частности, мы исследуем потенциал графовых нейронных сетей (GNNS) в выявлении паттернов вредного дискурса в социальных сетях. Наше исследование включает в себя создание набора данных онлайн-комментариев и сообщений, применение методов предварительной обработки, таких как токенизация, лемматизация и удаление стоп-слов, а также внедрение методов TF-IDF и встраивания слов для извлечения признаков. Мы используем подход, основанный на GNN, для классификации вредоносной речи. Это исследование имеет важное значение для повышения безопасности в Интернете и смягчения воздействия разжигания ненависти на общество.

***Ключевые слова**: онлайн социальные сети, машинное обучние, обработка естественного языка, сбор данных, деструктивная речь, анализ сетевого трафика.*

In the digital age, the spread of hate speech has become an urgent global problem, compounded by the rapid proliferation of content on online platforms and social media. This study examines advanced techniques for detecting destructive speech through the integration of natural language processing (NLP) and machine learning techniques. In particular, we are exploring the potential of graph neural networks (GNSS) in identifying patterns of harmful discourse on social media. Our research includes the creation of a dataset of online comments and posts, the application of preprocessing techniques such as tokenization, lemmatization, and removal of stop words, as well as the implementation of TF-IDF and word embedding techniques to extract features. We use a GNN-based approach to classify malicious speech. This research is important for improving Internet security and mitigating the impact of hate speech on society.

***Keywords:** online social networks, machine learning, natural language processing, data collection, destructive speech, network traffic analysis.*

Введение

В эпоху цифровых технологий распространение разжигающих ненависть высказываний стало одной из самых острых проблем, с которыми сталкиваются общества во всем мире [1, 2]. С развитием онлайн-платформ и социальных сетей вредоносная речь может распространяться с беспрецедентной скоростью, достигая огромной аудитории за считанные мгновения [3]. Разжигание ненависти, независимо от того, направлено ли оно против отдельных лиц или целых сообществ по признаку этнической принадлежности, религии, пола или других характеристик, может иметь серьезные социальные, психологические и политические последствия. В таких странах, как Казахстан, где проживает большое разнообразие этнических и культурных групп, разжигание ненависти может угрожать социальной гармонии и национальному единству.

Поскольку онлайн-коммуникации продолжают доминировать в повседневной жизни, традиционные методы модерации контента часто оказываются недостаточными для решения масштабной и сложной проблемы. Это привело к росту интереса к передовым технологиям, таким как системы обнаружения разжигающих ненависть высказываний и анализ сетевого трафика, которые предлагают новые способы мониторинга и смягчения последствий вредоносного поведения в Интернете. Обнаружение разжигающих ненависть высказываний использует машинное обучение и методы обработки естественного языка для выявления оскорбительного или вредоносного контента, в то время как анализ сетевого трафика позволяет отслеживать онлайн-активность и взаимодействия с целью выявления закономерностей, которые могут указывать на скоординированные кампании по разжиганию ненависти или распространение экстремистских идеологий.

В данной статье будет рассмотрены методы обнаружения разжигающих ненависть высказываний и анализа сетевого трафика. Цель исследования – оценить возможности использования графических нейронных сетей для выявления деструктивной речи в социальных сетях и предложить новые методы для повышения их точности и эффективности.

Научные методы, применяемые для определения разжигающих ненависть высказываний и анализа сетевого трафика

В контексте выявления разжигающих ненависть высказываний и анализа сетевого трафика могут использоваться различные методы и техники. Эти методы охватывают как обработку естественного языка (NLP), так и анализ сетевого трафика, каждый из которых играет решающую роль в выявлении и устранении вредоносного онлайн-контента.

Для начала рассмотрим методы обнаружения разжигающих ненависть высказываний, к ним относятся:

1. Методы машинного обучения (ML):

а) Контролируемое обучение – это наиболее распространенный метод, используемый для выявления высказываний, разжигающих ненависть. При контролируемом обучении для обучения моделей используются помеченные наборы данных, содержащие примеры высказываний, содержащих ненависть, и высказываний, не связанных с ненавистью. Такие алгоритмы, как логистическая регрессия, метод опорных векторов (SVM) и случайные леса, могут быть использованы для классификации текста как вызывающего ненависть или не вызывающего ненависти на основе этих помеченных примеров [4].

б) Более продвинутые методы, такие как рекуррентные нейронные сети (RNN), сети с длительной кратковременной памятью (LSTM) и сверточные нейронные сети (CNN), показали себя многообещающими в обнаружении сложных паттернов в тексте. BERT и его варианты (такие как RoBERTa или DistilBERT) очень эффективны для понимания контекста и языковых нюансов, что делает их полезными для выявления тонких форм разжигания ненависти, таких как сарказм или непрямые выражения [5, 6].

в) Комбинирование нескольких моделей или методик (например, сочетание дерева решений с методом опорных векторов) может помочь повысить эффективность классификации, снизить количество ошибок.

2. Методы обработки естественного языка (NLP)

а) Предварительная обработка текста. Перед применением моделей машинного обучения исходные текстовые данные очищаются и предварительно обрабатываются. Это включает в себя удаление стоп-слов, стемминг, лемматизацию и токенизацию, которые разбивают текст на более мелкие компоненты (такие как слова или фразы), которые легче поддаются анализу [7, 8].

б) Извлечение признаков. Различные признаки, такие как TF-IDF, вставки слов (например, Word2Vec, GloVe) и вставки предложений, извлекаются из текста для представления его значения в числовой форме. Эти функции помогают алгоритмам машинного обучения анализировать семантическую и синтаксическую структуру контента [9, 10].

в) Анализ настроений. Анализ настроений можно использовать для определения тона текста (позитивный, негативный, нейтральный). Ненавистнические высказывания часто имеют ярко выраженный негативный оттенок, и использование анализа настроений может помочь улучшить системы обнаружения.

3. Подходы, основанные на лексиконе

а) Лексикон ненавистнических высказываний. Более простой подход предполагает использование заранее определенных списков или словарей оскорбительных слов или фраз, которые, как известно, ассоциируются с ненавистническими высказываниями. Эти словари можно использовать для

фильтрации и маркировки контента, хотя этот метод часто не так детализирован, как методы, основанные на машинном обучении, поскольку он может упускать контекст или тонкие формы разжигания ненависти.

б) Контекстуальная лексика. Некоторые системы сочетают методы, основанные на лексиконе, с контекстно-зависимыми алгоритмами, которые учитывают окружающие слова и фразы, помогая определить, используется ли конкретное слово вредным образом.

в) Многоязычные подходы. Поскольку ненавистнические высказывания не ограничены каким-либо одним языком, разработка систем, способных обнаруживать ненавистнические высказывания на нескольких языках, имеет жизненно важное значение. Это может включать в себя обучение многоязычных моделей или адаптацию моделей для распознавания вариаций ненавистнических высказываний в различных языковых и культурных контекстах.

В ряд методов анализа сетевого трафика можно отнести:

1. Мониторинг сетевого трафика и сбор данных. Эти методы включают перехват и анализ пакетов данных, передаваемых по сети. DPI позволяет тщательно проверять полезную нагрузку и заголовки данных, потенциально выявляя схемы вредоносного трафика, в том числе те, которые используются для распространения разжигающих ненависть высказываний или экстремистского контента [11].

2. Анализ потока. Сетевой трафик также может быть проанализирован на уровне потока, где анализируются схемы взаимодействия между различными устройствами или системами. Аномалии в потоке трафика, такие как внезапный всплеск взаимодействия между учетными записями или IP-адресами, которые являются частью групп ненависти, могут быть отмечены для дальнейшего изучения [12].

3. Обнаружение аномалий трафика. Этот метод позволяет идентифицировать вредоносную активность на основе известных шаблонов или сигнатур. Например, если известно, что сеть используется для распространения разжигающих ненависть высказываний или экстремистских материалов, трафик из этих источников может быть помечен как подозрительный.

4. Анализ на основе графов

а) Анализ сетевого трафика может быть применен к платформам социальных сетей или форумам путем выявления связей и взаимодействий между пользователями. Анализируя поведение пользователей и модели общения, можно идентифицировать группы пользователей, которые используют разжигающие ненависть высказывания или другие вредоносные действия. Основанные на графах алгоритмы, такие как обнаружение сообщества, могут быть использованы для отслеживания скоординированных усилий по распространению разжигающих ненависть высказываний.

б) Боты часто играют важную роль в распространении разжигающих ненависть высказываний в Интернете. Анализ сетевого трафика может помочь выявить поведение, подобное поведению ботов, например, автоматические и частые публикации, и отметить такие аккаунты для дальнейшего изучения.

5. Фильтрация по ключевым словам и контенту:

а) Фильтрация URL-адресов и доменов. Подозрительный сетевой трафик может быть выявлен путем анализа URL-адресов и доменов, используемых для распространения материалов, содержащих ненависть. Отслеживая и блокируя URL-адреса, на которых регулярно размещаются или распространяются вредоносные материалы, анализ сетевого трафика может помочь снизить уровень распространения разжигающих ненависть высказываний.

б) Категоризация контента. Автоматизированные системы могут классифицировать и помечать контент в зависимости от его типа (например, текст, изображения, видео). Анализ типа публикуемого контента и схемы его распространения может выявить усилия, направленные на определенные группы с помощью риторики, наполненной ненавистью.

6. Системы обнаружения вторжений (IDS) и предотвращения вторжений (IPS). Эти системы отслеживают сетевой трафик на наличие признаков вредоносной активности, включая распространение вредоносного контента. Идентификаторы могут предупреждать администраторов о подозрительных проявлениях, связанных с разжиганием ненависти, в то время как IP-адреса могут активно блокировать такой трафик в режиме реального времени [13, 14].

7. Машинное обучение в анализе сетевого трафика. Модели машинного обучения могут использоваться для анализа огромных объемов данных о сетевом трафике и выявления едва заметных аномалий, которые могут указывать на наличие вредоносного контента или поведения. Эти модели могут выявлять закономерности, указывающие на скоординированные кампании по разжиганию ненависти или активность ботнетов.

а) Кластеризация и классификация. Используя методы обучения без контроля, такие как кластеризация k-means или DBSCAN, можно группировать необычный или вредоносный трафик, что позволяет аналитикам безопасности выявлять группы действий, связанных с ненавистью, и реагировать на них.

Анализируя вышеперечисленные методы мы можем сочетать методы обнаружения ненавистнических высказываний и анализа сетевого трафика. Данная интеграция обнаружения ненавистнических высказываний и анализа сетевого трафика может обеспечить более целостное представление о вредоносной онлайн-активности. Например, выявление подозрительных схем трафика и соотнесение их с ненавистническими высказываниями, выявлен-

ными с помощью NLP, может помочь выявить скоординированные атаки или распространение ненавистнических высказываний ботами или организованными группами. Сочетание обоих методов может обеспечить мониторинг в режиме реального времени и немедленные действия по предотвращению распространения разжигающих ненависть высказываний до того, как они распространятся широко. Эти методы и технички могут работать вместе для создания надежной и всеобъемлющей системы обнаружения и устранения разжигающих ненависть высказываний как в контенте, так и в сетевом трафике.

Материалы и методы

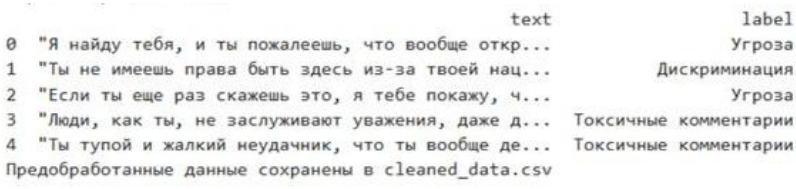
В данной части исследования авторы анализировали возможности использования графических нейронных сетей для выявления деструктивной речи в социальных сетях. С этой целью мы рассмотрим, как GNN используются для определения деструктивного контента, эффективно обрабатывая взаимосвязанные и социальные сети в текстах. Исследование состоит из нескольких этапов:

1. Сбор данных – это процесс сбора и систематизации информации, необходимой для конкретного исследования или анализа. Как правило, это означает формирование набора информации, содержащей данные, связанные с конкретной темой или вопросом. Сбор данных через социальные сети или другие источники – это процесс поиска необходимой информации, ее упорядочивания и подготовки к дальнейшей обработке и анализу. Для сбора данных из выбранных источников используются различные инструменты и методы, включая получение данных с помощью автоматизированных систем и программ (API) или возможность сбора данных вручную пользователем. Датасет, использованный в этом исследовании, состоит из комментариев и сообщений, опубликованных в различных социальных сетях. В ходе сбора датасета было отобрано большое количество текстов и типов контента, что было важно для понимания различных типов деструктивной речи в социальных сетях. Каждая запись или мнение взяты из определенного контекста, поэтому в них отражаются мнения и эмоции разных авторов. В процессе сбора данных тексты отбирались по специальным критериям. Процесс сбора был автоматизирован и систематизирован, что позволяло обрабатывать большие объемы информации. Поскольку собранные данные были получены с разных периодов времени, стало возможным также изучить тенденции в социальных сетях и то, как опасный контент меняется с течением времени. В целом датасет состоит из записей разных пользователей на разных социальных платформах и направлен на выявление в них особенностей деструктивной речи. Все данные были предварительно обработаны и подготовлены в соответствии с основными показателями и признаками, необходимыми для исследования. Собранный датасет состоит из 4000 строк и 4 классов.

2. Проведение препроцессинга. Препроцессинг (или предварительная обработка) – это процесс обработки данных перед их вводом в модели анализа или машинного обучения. На этом этапе для улучшения качества данных используются различные методы, такие как обработка потерянных или ошибочно введенных данных, удаление избыточных или повторяющихся данных. При работе с текстовыми данными могут использоваться такие методы, как разделение их на токены, приведение значений в стандартный формат и удаление стоп слов:

- Чтение .CSV файла: чтение данных из файла dataset.csv.
- Пропущенные строки: строки с пустыми значениями по столбцу "text" удаляются.
- Изменение регистра текстов: тексты преобразуются в строчные буквы.
- Удаление стоп-слов: используя библиотеку "stopwords" удаляются стоп-слова (например, "тоже", "или").
- Лемматизация: каждое слово приводится к своему корню (лемме).
- Данные сохраняются в новый файл cleaned_data.csv.

Также важно сбалансировать диапазон различных данных путем масштабирования и нормализации данных. В результате правильной обработки данных в первую очередь модели могут работать более эффективно, что повышает точность результатов исследования или анализа (рис. 1).



	text	label
0	"Я найду тебя, и ты пожалеешь, что вообще откр...	Угроза
1	"Ты не имеешь права быть здесь из-за твоей нац...	Дискриминация
2	"Если ты еще раз скажешь это, я тебе покажу, ч...	Угроза
3	"Люди, как ты, не заслуживают уважения, даже д...	Токсичные комментарии
4	"Ты тупой и жалкий неудачник, что ты вообще де...	Токсичные комментарии

Предобработанные данные сохранены в cleaned_data.csv

Рис. 1. Пример собранного датасета для выявления обнаружения ненавистнических высказываний

3. Визуализация данных. WordCloud (облако слов) – это метод визуального отображения наиболее часто встречающихся слов в тексте, при котором объем слов пропорционален их частоте. Такой подход помогает быстро определить основные термины и темы в тексте. Обычно слова, которые встречаются чаще, отображаются большим шрифтом, а слова, которые встречаются реже, меньшим. WordCloud особенно удобен для анализа текстов, проведения маркетинговых исследований и использования в научных работах.

Для извлечения цифровых данных из текстов были использованы методы TF-IDF и Bag of Words. При запуске кода читается текст из набора данных. Затем методом TF-IDF из текстов извлекаются слова и вычисляются их TF-IDF значения, после чего эти значения выводятся в виде матрицы. Точно так же с помощью метода Bag of Words частоты слов в текстах вычисляются и отображаются в виде матрицы. Для каждого метода полученные слова и матрицы выводятся на экран (рис. 2).

Рис. 2. TF-IDF матрица собранного датасета

4. Обучение через графические нейронные сети. Нейронные сети графов (Graph Neural Networks, GNNs) – это нейронные сети, предназначенные для обработки данных, расположенных в структуре графов. Эти сети позволяют анализировать информацию, связанную в виде графов. Графические нейронные сети формируют новое представление каждого узла, получая информацию от своих соседей. Технология GNN эффективно используется в различных областях, особенно когда важны связи и отношения.

Результаты исследования

Как уже говорилось выше, сначала данные извлекаются из файла .CSV, а затем тексты преобразуются в числовые векторы с помощью метода TF - IDF. Это определяет косинусное сходство между векторами и создает график, показывающий связи между похожими текстами. Узлы графа представляют тексты, а ребра основаны на косинусном сходстве. Затем Граф NetworkX преобразуется в формат PyTorch Geometric, где особенности узлов обозначаются векторами TF-IDF, а ребра представляют связи графа. Далее используется нейронная сеть GSN. Модель использует GCNConv

(Graph Convolutional Layer), состоящий из двух слоев. После первого слоя используется функция активации ReLU. Вход модели – это характеристики узлов, а выход – логиты, которые можно использовать для прогнозирования классов.

Для обучения этой модели мы использовали crossentropyloss, который используется в задачах классификации. Adam используется в качестве оптимизатора для тренировки модели. На каждом этапе вычисляются выходы и потери модели, а затем обновляются настройки. Обучение состоит из 200 эпох, но loss печатается через каждые 10 эпох (рис. 3).

```
Epoch 0, Loss: 0.6935852766036987
Epoch 10, Loss: 0.4194245934486389
Epoch 20, Loss: 0.2822166085243225
Epoch 30, Loss: 0.23113419115543365
Epoch 40, Loss: 0.20417654514312744
Epoch 50, Loss: 0.1855730265378952
Epoch 60, Loss: 0.17114481329917908
Epoch 70, Loss: 0.15963691473007202
Epoch 80, Loss: 0.15078242123126984
Epoch 90, Loss: 0.14198096096515656
Epoch 100, Loss: 0.13467492163181305
Epoch 110, Loss: 0.1286131888628006
Epoch 120, Loss: 0.1232755109667778
Epoch 130, Loss: 0.11773569881916046
Epoch 140, Loss: 0.1127094253897667
Epoch 150, Loss: 0.10859856754541397
Epoch 160, Loss: 0.10483598709106445
Epoch 170, Loss: 0.10144753009080887
Epoch 180, Loss: 0.09834213554859161
Epoch 190, Loss: 0.09540204703807831
```

Рис. 3. loss значения при обучении модели

Далее измеряется точность модели. Для этого сначала модель переводится в режим оценки (evaluation). Затем по всем данным рассчитываются расходы модели. Для каждого узла делается предположение, выбирая класс с наибольшей вероятностью. Для проверки тестовых данных используется маска с именем test_mask. По тестовым данным вычисляется точность, т.е. соответствие между прогнозом и истинными значениями (рис. 4).

Затем вычисляются Accuracy, precision, recall и F1-score тестовых данных. Эти показатели помогают оценить общую эффективность модели. Метрики рассчитываются с помощью параметра "weighted", который позволяет пропорционально добавлять баллы для каждого класса. Точность модели составила 96%.

```python
from sklearn.metrics import precision_score, recall_score, f1_score
# Оценка модели на тестовых данных
model.eval()  # Устанавливаем модель в режим оценки
# Получаем предсказания на всех данных
out = model(graph_data.x, graph_data.edge_index)
# Получаем классы с максимальной вероятностью для каждого узла
pred = out.argmax(dim=1)
# Выбираем только тестовые данные (используя маску test_mask)
test_preds = pred[test_mask]
test_labels = graph_data.y[test_mask]
# Выводим точность на тестовых данных
accuracy = (test_preds == test_labels).sum().item() / len(test_labels)
print(f'Accuracy on test data: {accuracy:.4f}')
# Преобразуем предсказания в массивы для метрик
test_preds = test_preds.cpu().numpy()
test_labels = test_labels.cpu().numpy()
# Точность
precision = precision_score(test_labels, test_preds, average='weighted')
recall = recall_score(test_labels, test_preds, average='weighted')
f1 = f1_score(test_labels, test_preds, average='weighted')

print(f'Precision: {precision:.4f}')
print(f'Recall: {recall:.4f}')
print(f'F1-score: {f1:.4f}')
```

```
Accuracy on test data: 0.9688
Precision: 0.9693
Recall: 0.9688
F1-score: 0.9688
```

Рис. 4. Этап измерения точности модели

Заключение

Использование графических нейронных сетей для выявления деструктивной речи в социальных сетях – одна из важнейших областей современной обработки естественного языка и машинного обучения. В этом исследовании мы в первую очередь собрали соответствующий датасет и провели работу по очистке текстов, лемматизации и приведению слов в нижний регистр на этапе предварительной обработки. В результате анализа мы использовали генерацию Word Cloud для извлечения важных функций из текстов, что позволило нам сосредоточиться на наиболее часто используемых словах и терминах в текстах.

На следующем этапе было проведено обучение моделям с помощью графических нейронных сетей. Благодаря использованию графических нейронных сетей стало возможным получать высокоточные результаты классификации, учитывающие отношения в социальных сетях, связи между пользователями и контекст.

В целом, в результате этого исследования были открыты новые возможности в области выявления деструктивной речи в социальных сетях путем демонстрации эффективности графических моделей. Эти методы в будущем внесут важный вклад в раннее выявление опасной речи в онлайн-среде и принятие решений по ее управлению.

Данная работа была выполнена в рамках проекта, финансируемого Комитетом Науки Министерства науки и высшего образования Республики Казахстан (грант AP19576868, руководитель проекта Болатбек М.А.).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Ahmed El-Sayed and Omar Nasr.* AAST-NLP at Multimodal Hate Speech Event Detection 2024: A Multimodal Approach for Classification of Text-Embedded Images Based on CLIP and BERT-Based Models // In Proceedings of the 7th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2024). St. Julians, Malta. Association for Computational Linguistics. 2024. – P. 139-144.
2. *Raturi A., Joshi K., Anupriya, Jain P., Gupta V.K. and Meena J.* "Hate Speech Detection System using Machine Learning Algorithms // 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, 2024. – P. 446-451. – DOI: 10.1109/InCACCT61598.2024.10551015.
3. *Al-Saqqa, Samar & Awajan, Arafat & Hammo, Bassam.* A Survey of Hate Speech Detection for Arabic Social Media: Methods and Datasets // Procedia Computer Science. – 2024. – 251. – P. 224-231. – 10.1016/j.procs.2024.11.104.
4. *Abro Sindhu & Shaikh Sarang & Hussain Zahid & Ali Zafar & Khan, Sajid & Mujtaba Ghulam.* Automatic Hate Speech Detection using Machine Learning: A Comparative Study // International Journal of Advanced Computer Science and Applications. – 2020. – 11. – 10.14569/IJACSA.2020.0110861.
5. *Selam Abitte Kanta, Grigori Sidorov, and Alexander Gelbukh*. Selam@DravidianLangTech 2024:Identifying Hate Speech and Offensive Language // In Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages. St. Julian's, Malta. 2024. – P. 91-95.
6. *Iorliam Aamo & Agber Selumun & Dzungwe MP & Kwaghtyo DK & Bum Sylvester*. Comparative Analysis of Deep Learning Techniques for the Classification of Hate Speech // NIGERIAN ANNALS OF PURE AND APPLIED SCIENCES. – 2021. – 4. – P. 121-128. – 10.46912/napas.227.
7. *Ilhan Syed & Soubraylu Sivakumar & Nagaraj Jayanth & Ramesh S. & Sreeram N. & Rajalakshmi Ratnavel*. Hate Speech Detection and Classification Using NLP. – 2024. – P. 1-7. – 10.1109/ICAIT61638.2024.10690655.
8. *Shohan, Mehedi Hasan et al.* Use of Natural Language Processing for the Detection of Hate Speech on Social Media // Journal of Advanced Research in Applied Sciences and Engineering Technology. – (2024: n. pag.
9. *Putri Mila & Setiawan Erwin.* Feature Expansion Using Word2vec for Hate Speech Detection on Indonesian Twitter with Classification Using SVM and Random Forest. – 2022. – 10.30865/mib.v6i2.3855.

10. *Dalavi S., Nivelkar T., Patil S., Sawant A. and Aylani A.* Comparative Analysis of Vectorization Techniques and Machine Learning Models for Hate Speech Detection // 2023 Global Conference on Information Technologies and Communications (GCITC), Bangalore, India, 2023. – P. 1-5. – doi: 10.1109/GCITC60406.2023.10426214.
11. *Jakkani Anil*. Real-Time Network Traffic Analysis and Anomaly Detection to Enhance Network Security and Performance: Machine Learning Approaches // Journal of Electronics Computer Networking and Applied Mathematics. – 2024. – 4. – P. 2799-1156. –10.55529/jecnam.44.32.44.
12. *Dommeti, Dinesh*. Network traffic analysis and alerting system // South Asian Journal of Engineering and Technology. – 2023. – 13. – P. 1-9. – 10.26524/sajet.2023.10.
13. *Abbas Safana & Naser Wedad & Abbas Amal*. Subject review: Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) // Global Journal of Engineering and Technology Advances. – 2023. – 14. – P. 155-158. – 10.30574/gjeta.2023.14.2.0031.
14. *Akwasi Adu-Kyere, Ethiopia Nigussie, Jouni Isoaho.* Analyzing the effectiveness of IDS/IPS in real-time with a custom in-vehicle design // Procedia Computer Science. – 2024. – Vol. 238. – P. 175-183. – ISSN 1877-0509. – https://doi.org/10.1016/j.procs.2024.06.013.

УДК 004.056

М.О. Калинин, А.С. Коноплев, Е.В. Завадский, Г.С. Кубрин

Санкт-Петербургский политехнический университет Петра Великого,
Россия, г. Санкт-Петербург

БЕЗОПАСНОСТЬ БЛОКЧЕЙН-СЕТИ УМНОГО ГОРОДА: ЗАЩИТА ОТ ЭГОИСТИЧНОГО МАЙНИНГА

Представлен метод защиты блокчейн-сети от эгоистичного майнинга. Исследован алгоритм майнинга в блокчейнах с механизмом консенсуса Proof-of-Work. Представлено решение для защиты от эгоистичного майнинга, позволяющее анализировать закономерности данных, поступающих из майнинг-пула, идентифицировать атакующий пул и достичь низкой доли ложноположительных и ложноотрицательных результатов.

Ключевые слова: *блокчейн, безопасность, пул, умный город, эгоистичный майнинг.*

A method for protecting a blockchain network from selfish mining is presented. A mining algorithm in blockchains with a proof-of-work consensus mechanism is analyzed. A solution for protecting against selfish mining is presented, allowing for the analysis of patterns in data coming from a mining pool, identification of attacking pools, and achievement of a low rate of false positives and false negatives.

Keywords: *blockchain, security, pool, smart city, selfish mining.*

Введение

Умный город – эффективная технологическая концепция, которая обеспечивает повышение качества жизни в современной городской инфраструктуре за счет оптимального управления информацией и физическими процессами [1]. Критические компоненты «умного города», такие как Интернет вещей, промышленные коммуникации, сенсорные сети, транспорт, энергосистемы и электронные услуги (например, ситуативное управление, медобслуживание, электронное голосование и пр.), работают в режиме реального времени, обеспечивая комфорт проживания и непрерывность бизнеса.

Одновременно с этим развитие технологии «умного города» сталкивается с актуальными проблемами кибербезопасности [2, 3]. Среда умного города производит большие объемы данных, генерируемых разными источниками. Технология блокчейн является одним из механизмов, который позволит решить данные проблемы безопасности данных [4, 5].

Блокчейн-сети «умных городов» обеспечивают возможность хранения и обмена конфиденциальной информацией, исключая недоверенных посредников и обеспечивая децентрализацию архитектуры системы умного города [6–9]. Процесс создания новых блоков и решения хэш-функции называется майнингом. Участников майнинга называют майнерами. Майнеры получают вознаграждение за свою работу от сети. Атака эгоистичного майнинга – атака на блокчейн, при которой майнеры скрывают недавно обнаруженные блоки вместо того, чтобы публиковать их для остальной части сети [10]. Эгоистичные майнеры продолжают майнить на своей частной цепочке, в то время как обычные майнеры тратят ресурсы на майнинг на более короткой цепочке. Согласно протоколу блокчейна, более длинная цепочка имеет приоритет, а короткие цепочки отбрасываются. Это позволяет эгоистичным майнерам получить преимущество, сохраняя свою цепочку в секрете. Если эгоистичный майнер находит более одного блока раньше кого-либо другого, это может заставить других майнеров тратить время, работая над неправильной цепочкой. Когда эгоистичный майнер приближается к победе, он раскрывает свою секретную цепочку, и работа всех остальных майнеров, таким образом, становится ненужной, поскольку сеть использует самую длинную цепочку [15]. Эта атака может использоваться злоумышленниками для получения непропорционально большого вознаграждения за майнинг или (в сочетании с другими атаками) для нарушения работы блокчейн-сети.

Целью представленного исследования является выявление и блокировка майнинг-пула, осуществляющего атаку эгоистичного майнинга.

Анализ существующих решений

Для противодействия угрозе эгоистичного майнинга в настоящее время разработано около 20 методов защиты, которые представляют собой три основных подхода: анализ высоты форков блокчейна; применение машинного обучения, которое контролирует различные особенности форков блокчейна; анализ атрибутов транзакций, а не форков (табл. 1).

Таблица 1

Существующие подходы к выявлению эгоистичного майнинга

Подход	Точность результатов, %	Критика
Анализ высоты форков [12, 13]	95-98	Невозможно определить атакующего. Доля ложноотрицательных результатов (FN): 14%
Применение машинного обучения [14–17]	98	Невозможно определить атакующего. Зависимость от наличия набора данных и качества обучения. FN: 23,7–24,4%
Анализ атрибутов транзакций [18–20]	99	Невозможно определить атакующего. Зависимость от наличия набора данных и качества обучения. FN=1,5–4%

Основной недостаток известных методов заключается в том, что они потенциально могут обнаружить атаку эгоистичного майнинга, но не могут предотвратить ее в реальном масштабе времени. Некоторые методы сталкиваются с техническими проблемами реализации – например, большинство известных решений требуют обновления существующих протоколов, что видится сложным и затратным. Также они требуют привязки криптоадресов к пулу майнинга, что может быть невыполнимо в реальных системах. Все рассмотренные детекторы эгоистичного майнинга не способны идентифицировать конкретный пул эгоистичного майнинга. Кроме того, большинство существующих методов показало высокий уровень ложноотрицательных срабатываний (false negative, FN) до 24,4%.

Разработка метода защиты от эгоистичного майнинга

Крупномасштабные системы «умного города», построенные с использованием блокчейн-сети с высокой капитализацией, привлекательны для эгоистичных майнеров. По мере усложнения блокчейн-сетей одиночный майнинг перестал приносить доход майнерам, и майнеры объединяются в пулы. Вычислительная мощность, распределенная между майнинг-пулами, значительно превышает мощность отдельных майнеров. Чем ниже капитализация блокчейн-сети, тем меньшую долю мощности хэширования блокчейна занимают майнинг-пулы.

Теоретическая оценка эффективности эгоистичной стратегии майнинга представлена в [21]. Уравнение описывает ожидаемый доход эгоистичного майнера в зависимости от процентной доли в мощности сети и того, какая часть честного сообщества работает над цепочкой, полученной от эгоистичных майнеров:

$$R_{pool} = \frac{r_{pool}}{r_{pool} + r_{others}} = \frac{\alpha(1-\alpha)^2(4\alpha + \gamma(1-2\alpha)) - \alpha^3}{1-\alpha(1+(2-\alpha)\alpha)},$$

где R_{pool} – отношение блоков, добытых нечестной шахтой, ко всем добытым блокам; r_{pool} – доход эгоистичных майнеров; r_{others} – доход остальной сети; α – доля мощности майнинга от общей мощности сети; $(1-\alpha)$– остальная часть мощности сети, γ – часть честных узлов, которые решили работать в цепочке, происходящей в результате нечестного майнинга, $(1-\gamma)$ – оставшаяся часть честных майнеров, выполняющих работу по версии цепочки, исходящей от честных узлов.

Согласно [21], для параметра α (доли мощности майнинг-пула в общей мощности сети) при $\alpha = \{0.25, 0.30\}$ наблюдается прямая связь между выходной мощностью и количеством извлеченных блоков. Далее с ростом

α эта зависимость становится логарифмической. Это связано с увеличением количества форков блокчейна в результате меньшей сложности извлечения блока и возросшей мощности сети. Таким образом, для эгоистичного майнера майнинг имеет смысл только при наличии достаточной мощности, в противном случае расходы эгоистичного майнера становятся непропорционально высокими по сравнению с вознаграждением. Поэтому для обнаружения атаки следует сосредоточиться на анализе поведения майнинг-пулов, так как атака эгоистичного майнинга приносит пользу участникам с хэшрейтом, который намного больше, чем у других.

Для анализа поведения майнинг-пула необходимо контролировать майнинг. Это включает в себя выбор значения для определенного параметра *nonce*, что позволяет получить хэш для текущего блока. Майнер подключается и общается с майнинг-пулом с помощью специального протокола *Stratum V2* [22] (для алгоритма консенсуса Proof-of-Work). Протокол *Stratum V2* использует архитектуру клиент-сервер, где майнер подключается к пулу, инициализируется и начинает получать от него задания. В контексте атаки эгоистичного майнинга полезно проанализировать, как задания получаются от майнинг-пула, называемого *share* (рис. 1). *Share* содержит *coinbase*-транзакцию, в которой указан адрес майнинг-пула для получения вознаграждения. Когда майнер находит решение (*share*), он отправляет его в пул для проверки. В случае успеха решение используется в качестве решения для основной задачи сети блокчейна.

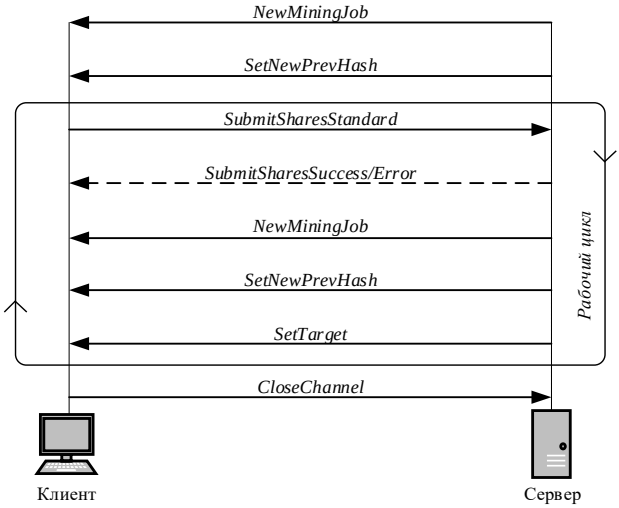


Рис. 1. Схема получения заданий (согласно Stratum V2)

Майнинг-пул может выполнить атаку эгоистичного майнинга, используя мощность майнеров, подключенных к серверу, согласно представленному сценарию в соответствии с протоколом *Stratum V2*:

1) Майнинг-пул отправляет задачи майнерам с помощью *NewMiningJob* и устанавливает *NewPrevHash* (*prevhash*(*n*)).

2) Когда решение совпадает с решением для основной задачи блокчейна *SubmitSharesSuccess* (*prevhash*(*n*++)), *SetNewPrevHash* устанавливает *prevhash*(*n*++).

3) Майнинг-пул отправляет задачи майнерам с помощью *NewMiningJob*, а *SetewPrevHash* устанавливает *prevhash*(*n*++).

4) Шаги 1–3 повторяются, пока кто-то другой не обнаружит решение для блока блокчейна.

5) *n* решений блоков публикуются одновременно, и майнер получает большее вознаграждение, так как выбирается более длинная цепочка.

Если майнер работает внутри майнинг-пула, он получает шаблоны заданий и не может изменить адрес криптокошелька, на который отправляется вознаграждение за добытый блок. Поэтому атака эгоистичного майнинга может быть осуществлена только майнинг-пулом.

Разработанный метод основан на вышеизложенной концепции и реализован в виде клиентского плагина для программного обеспечения майнера.

Майнер использует специальное программное обеспечение для подключения к майнинг-пулу и выполнения алгоритмов консенсуса Proof-of-Work. На рис. 2 показана схема работы предлагаемого решения. Прототип разработан на языке *Python* в виде плагина для *CGminer* v.3.7.2 – популярной утилиты майнеров с открытым исходным кодом GPU/FPGA/ASIC.

Построенное решение подключается к *CGminer* через сокеты. Затем функция вызова применяется для передачи данных в плагин. В ответ получается *NewPrevHash* от *CGminer*. Если значение *NewPrevHash* не найдено, то решение отправляет код ошибки в *CGMiner*, который в свою очередь сигнализирует о том, что майнер скрывает предыдущий блок и пытается создать скрытый форк блокчейна. Наконец, решение передает хеш-информацию ложного блока *PrevHash* и адрес криптокошелька на сервер, и каждый пользователь плагина знает, что блок с этим *PrevHash* находится в эгоистичной цепочке. Таким образом, анализ хэша предыдущего блока позволяет определить, скрывает ли майнинг-пул блок от сети или нет. В случае невалидности данных плагин предупреждает другие пулы и прерывает решение блоков в эгоистичной цепочке, блокируя принятие шаблонов по протоколу Stratum V2, лишая эгоистичного майнера вознаграждения и тем самым сокращая вознаграждение эгоистичного майнера.

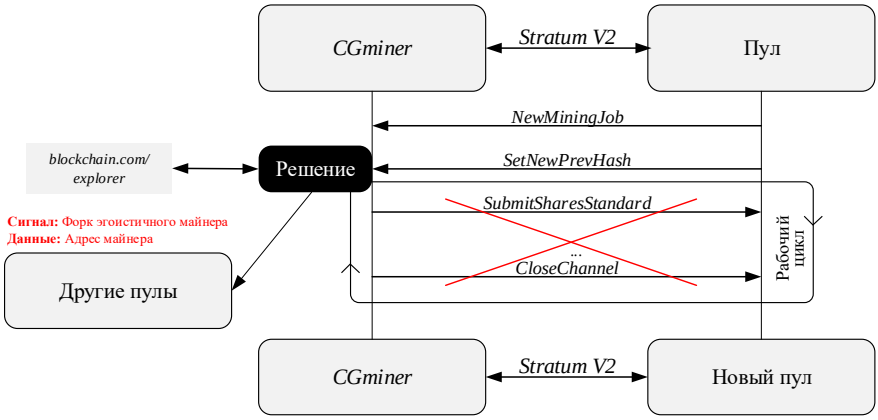


Рис. 2. Схема разработанного решения

Для экспериментальной оценки разработанного метода построена программная модель блокчейна с механизмом консенсуса Proof-of-Work. Сгенерировано 150000 образцов цепочек. Генерация альтернативных форков и эгоистичного майнинга выполнена с использованием конечного автомата [23].

Разработанный метод демонстрирует преимущество в полноте обнаружения атак эгоистичного майнинга, идентифицируя атакующий пул и предотвращая использование возможностей участников пула для реализации эгоистичного майнинга. Решение имеет более высокий уровень точности и низкий уровень ложных срабатываний, чем у конкурирующих методов. Согласно экспериментам, разработанный плагин имеет точность 98,7% и уровень ложноположительных и ложноотрицательных результатов 1,2% и 5,1%, соответственно.

Заключение

Построенное решение может быть интегрировано в программное обеспечение майнера и уменьшить негативное влияние атак эгоистичного майнинга на критические сети, использующие блокчейн, такие как умные города, Интернет вещей, метавселенные, интеллектуальные производства, финансовые и контракт-ориентированные системы.

Поскольку блокчейн может испытывать задержки при публикации хешей для блоков из-за больших размеров блокчейн-сетей и сетевых задержек, следует учитывать параметр задержки и адаптивно пересчитывать порог задержки во время обнаружения эгоистичного майнинга, что составляет предмет дальнейшего исследования и развития предложенного решения.

Исследование выполнено за счет гранта Российского научного фонда №24-11-20005, https://rscf.ru/project/24-11-20005/, грант Санкт-Петербургского научного фонда (договор №24-11-20005 о предоставлении регионального гранта).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Kirimtat, Krejcar O., Kertesz A., Tasgetiren M.F.* Future trends and current state of smart city concepts: A survey// IEEE Access. – 2020. – 8. – P. 86448-86467.
2. *Ibrahim Khalil M.W., Kamel M.A.E.* Towards smart sustainable cities vision and challenges // International Journal of Nonlinear Analysis and Applications. – 2024. – 15 (3). – P. 261-274.
3. *Waseem Anwar R., Ali S.* Smart cities security threat landscape: A review // Computing and Informatics. – 2022. – 41. – P. 405-423.
4. *Sharma S., Mishra N.* Horizoning recent trends in the security of smart cities: Exploratory analysis using latent semantic analysis // Journal of Intelligent and Fuzzy Systems. 2024. – 46. – P. 579-596.
5. *Biswas S., Yao Z., Yan L., Alqhatani A., Bairagi A.K., Asiri F., Masud M.* Interoperability benefits and challenges in smart city services: Blockchain as a solution // Electronics. – 2023. – 12. – Paper number 12041036.
6. *Makani S., Pittala R., Alsayed E., Aloqaily M., Jararweh Y.* A survey of blockchain applications in sustainable and smart cities // Cluster Computing. – 2022. – 25. – P. 3915-3936.
7. *lnahari M.S., Ariaratnam S.T.* The application of blockchain technology to smart city infrastructure // Smart Cities. – 2022. – 5. – P. 979-993.
8. *Hakak S., Khan W.Z., Gilkar G.A., Imran M., Guizani N.* Securing smart cities through blockchain technology: Architecture, requirements, and challenges // IEEE Network. – 2020. – 34. – P. 8-14.
9. *Khalil U., Mueen-Uddin, Malik O.A., Hussain S.* A blockchain footprint for authentication of IoT-enabled smart devices in smart cities: State-of-the-art advancements, challenges and future research directions // IEEE Access. – 2022. – 10. – P. 76805-76823.
10. *Eyal, Sirer E.G.* Majority Is Not Enough: Bitcoin mining is vulnerable // Communications of the ACM. – 2018. – 61 (7). – P. 95-102.
11. *Madhushanie, Vidanagamachchi S., Arachchilage N.* Selfish mining attack in blockchain: A systematic literature review // International Journal of Information Security. – 2024. – 23. – P. 2333-2351.
12. *Kang H., Chang X., Yang R., Misic J., Misic V.B.* Understanding selfish mining in imperfect Bitcoin and Ethereum networks with extended forks // IEEE Transactions on Network and Service Management. 2021. – 18(3). – P. 3079-3091.
13. *Saad M., Njilla L., Kamhoua C., Mohaisen A.* Countering selfish mining in blockchains // International Conference on Computing, Networking and Communications. – 2019. – P. 360-364.
14. *Peterson M., Andel T., Benton R.* Towards detection of selfish mining using machine learning // International Conference on Cyber Warfare and Security. – 2022. – 17. – P. 237-243.
15. *Wang Z., Lv Q., Lu Z., Wang Y., Yue S.* ForkDec: Accurate Detection for Selfish Mining Attacks // Security and Communication Networks. – 2021. – 2021.

16. *Chicarino V., Albuquerque C., Jesus E., Rocha A.* On the detection of selfish mining and stalker attacks in blockchain networks // Annals of Telecommunications. – 2020. – 75 (3-4). – P. 143-152.
17. *Khan M.I.* Deep reinforcement learning for selfish nodes detection in a blockchain // French Regional Conference on Complex Systems. – 2023.
18. *Ritz F., Zugenmaier A.* The Impact of Uncle Rewards on Selfish Mining in Ethereum // IEEE European Symposium on Security and Privacy Workshops. – 2018. – P. 50-57.
19. *Tosh D.K., Shetty S., Liang X., Kamhoua C.A., Kwiat K.A., Njilla L.* Security implications of blockchain cloud with analysis of block withholding attack // IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, – 2017. – P. 458-467.
20. *Nikhalat-Jahromi, Saghiri A.M., Meybodi M.R.* Vdhla: Variable depth hybrid learning automatonits application to defense against the selfish mining attack in bitcoin // arXiv preprint arXiv:2302.12096. – 2023.
21. *Kędziora M., Kozłowski P., Szczepanik M., Jóźwiak P.* Analysis of Blockchain Selfish Mining Attacks // Advances in Intelligent Systems and Computing. – 2020. – 1050. – P. 231-240.
22. Past and future of bitcoin mining protocols: Stratum V2 overview. – Режим доступа: https://braiins.com/blog/past-and-future-of-bitcoin-mining-protocols-stratum-v2-overview.
23. *Lee S., Kim S.* Rethinking selfish mining under pooled mining // ICT Express. – 2023. – 9 (3). – P. 356-361.

УДК 004.056

В.О. Корнеев, Д.А. Волколупов, Е.Ю. Городецкая

Сибирский государственный университет науки и технологий,
Россия, г. Красноярск

МЕТОДИКА ФОРМИРОВАНИЯ ЗАДАНИЯ НА КИБЕРУЧЕНИЯ ЧЕРЕЗ АНАЛИЗ СВЯЗЕЙ УЯЗВИМЫХ УЗЛОВ

В статье рассматривается проблема автоматизированного формирования целевых групп для задач повышения осведомленности. Рассмотрен конкретный способ повышения осведомленности – проведение учебно-тренировочных занятий. Сформирован алгоритм подготовки задания для учебно-тренировочного занятия и процесс его подготовки, проанализирован подход и ограничения к выбору целевых групп. Предложена возможность работы с целевой группой через формирование гиперграфа, содержащего информацию об уязвимостях и условиях их реализации, а также критичных узлах, подверженных реализации угроз безопасности информации. Формирование гиперграфа позволит автоматизировать сопоставление целевой группы сотрудников для повышения осведомленности с конкретными узлами и уязвимостями. Применимо также для актуализации заданий на киберучения с учетом новых и модифицированных техник и тактик злоумышленников. В статье также приведен пример возможного использования подхода с применением киберполигона и конструктора сценариев.

Ключевые слова: *информационная безопасность, гиперграфы, инциденты ИБ, анализ угроз, киберучения.*

The article discusses the problem of automated formation of target groups for awareness-raising tasks. A specific way of raising awareness is considered – conducting educational and training sessions. The algorithm for preparing assignments for the training session and the process of its preparation are formed, the approach and limitations to the selection of target groups are analyzed. The possibility of working with the target group is proposed through the formation of a hypergraph containing information about vulnerabilities and their implementation conditions, as well as critical nodes exposed to information security threats. Creating a hypergraph will automate the matching of a target group of employees to increase awareness of specific nodes and vulnerabilities. It is also applicable for updating cyber-learning tasks, taking into account new and modified techniques and tactics of intruders. The article also provides an example of a possible use of a cyberpolygon approach and a script designer.

Keywords: *information security, hypergraphs, information security incidents, threat analysis, cyber studies.*

Введение

Активное внедрение информационных технологий обуславливает возникновение новых, ранее не существовавших, угроз безопасности информации, расширяя спектр потенциально возможных компьютерных атак.

Сегодня злоумышленники атакуют не только инфраструктуру организаций, но и рядовых сотрудников. Современные атаки становятся все более сложными и изощренными, и хакеры постоянно разрабатывают новые методы для достижения своих целей, против которых обычные пользователи особенно уязвимы. В таких условиях цифровая гигиена становится необходимым навыком для каждого сотрудника, особенно если речь идет о критической информационной инфраструктуре (далее – КИИ).

По данным отчета компании Positive Technologies об изменении ландшафта угроз в период с 2022 по 2024 года, фиксируется значительный рост атак с использованием шпионского ПО, доля которого выросла с 10% в 2022 году до 49% в 2024 г. Кроме этого наблюдается рост и других типов ВПО [1]. Подобные атаки стали активнее развиваться не только потому, что осуществляется переход на отечественные средства защиты информации, но и потому что их удобно доставлять, посредством, например, фишинговой атаки.

Кроме того, по оценкам современных исследований, существует значимая часть области информационной безопасности, которая связана с человеческим фактором, например для разных типов систем [2, 3].

В рамках работы объектом исследования является процесс обучения сотрудников организации вопросам ИБ, посредством проведения обучения и закрепляющих навыки учебно-тренировочных занятий. Предметом исследования выступает подготовка процесса автоматизации обучения. Это исследование позволит разработать эффективные методы и подходы к обучению, которые помогут сотрудникам более адекватно реагировать на потенциальные угрозы и минимизировать риски, связанные с информационной безопасностью.

Описание подхода к формированию учебно-тренировочных занятий

Для сотрудников, которые являются профильными специалистами по информационной безопасности или сотрудников, которые непосредственно работают с инфраструктурой организации, обучение можно построить в формате киберучений.

В зависимости от сценария тренировок и задействованных технических средств, киберучения могут подразделяться на штабные, практические и гибридные. Каждый из указанных классов рассмотрен ниже.

Штабные киберучения по обнаружению, реагированию и противодействию компьютерным атакам относятся к теоретическим киберучениям. В рамках штабных киберучений участники решают предложенные задачи (сценарии), при этом не взаимодействуя с реальным оборудованием, информационными системами и средствами защиты; в результате решения задачи участники учений предлагают организационно-технические меры, позволяющие не допустить реализацию рассматриваемого негативного сценария или минимизировать его последствия. Штабные киберучения, как правило, направлены на отработку взаимодействия сотрудников субъекта КИИ, а также взаимодействия с регуляторами. Целевой аудиторией киберучений данного типа являются руководители и технические специалисты компании.

Практические киберучения по обнаружению, реагированию и противодействию компьютерным атакам подразумевают моделирование реальных ситуаций (с использованием моделируемой инфраструктуры субъекта КИИ, команды атакующих, средств защиты информации и т.д.) для последующей отработки навыков мониторинга, обнаружения и противодействия компьютерным атакам. В ходе киберучений участники будут отрабатывать сценарии, с реализацией которых они могут столкнуться в реальной жизни и на реальном оборудовании. Проведение киберучений данного типа позволяет участникам сформировать практические навыки, необходимые для реагирования на реальные инциденты в случае их возникновения.

Гибридные киберучения по обнаружению, реагированию и противодействию компьютерным атакам являются «объединением» двух описанных выше киберучений: сценарий таких киберучений прописан заранее и, как правило, содержит теоретическую часть, реализуемую в формате командно-штабной тренировки и практическую часть, в которой для демонстрации векторов атаки и действий атакующих воссоздается часть инфраструктуры либо с помощью реального оборудования, либо эта инфраструктура эмулируется с использованием специализированных программно-технических комплексов. В зависимости от уровня сложности сценария, участниками данного типа киберучений могут быть как представители топ-менеджмента, так и технические специалисты.

Автоматизировать такие мероприятия можно с помощью проведения киберучений на специализированных киберполигонах.

Процесс автоматизации киберучений

Процесс будет состоять из 11 функций и 4 отделов, в которых происходят действия, связанные с процессом.

Формирование бюллетеней безопасности:

Вход: данные об уязвимостях, актуальная информация о текущих угрозах и атаках.

Выход: сформированная, регулярно направляемая бюллетень, содержащая описание угрозы, IP- и URL-адреса, хеш-суммы.

Требования:

- актуальность и точность содержащейся информации;
- подробное описание уязвимостей для принятия решения сотрудниками отдела ИБ.

Выявление сотрудниками отдела ИБ уязвимостей, которые можно решить обучением

Вход: бюллетень информационной безопасности, направленный в организацию.

Выход: решение о возможности проведения киберучения.

Требования:

- должны быть выявлены уязвимости, к которым можно разработать учебные и тренировочные материалы, для реализации обучения сотрудников организации;
- уязвимости должны быть применимы к организации.

Запрос перечня сотрудников

Вход: решение о проведение обучения и тренировочных занятий среди сотрудников.

Выход: запрос перечня в отделе кадров.

Требования:

- отделом ИБ должно быть принято решение о проведении обучения сотрудников по выбранной уязвимости;
- определена целевая категория сотрудников для обучения.

Формирование перечня сотрудников

Вход: получение запроса о сотрудниках, относящихся к определённой категории.

Выход: передача информации о сотрудниках и подготовка служебных записок.

Требования:

- категория необходимых сотрудников однозначно определена и сформирована;
- указана численность, для которой возможно провести учебно-тренировочное занятие;
- указаны планируемые сроки проведения обучения и занятия.

Формирование служебной записки на обучение

Вход: перечень сотрудников, направленных на обучение.

Выход: уведомление сотрудников о необходимости прохождения обучения.

Требования:

- сотрудники должны удовлетворять требованиям, ранее заявленным сотрудниками отдела ИБ;
- сотрудники, которые будут проходить обучения, не должны пребывать в запланированные даты в отпуске;
- сотрудники должны быть заранее уведомлены о необходимости прохождения обучения.

Разработка обучающих материалов

Вход: получение информации от отдела кадров.

Выход: информирование сотрудников о необходимости прохождения обучения и подготовка тренировочного занятия.

Требования:

- обучающий материал, должен быть подобран согласно теме, определенной выше;
- обучающий материал, должен быть сформирован с учетом планируемой учебной группы, при этом он должен быть достаточен.

Разработка учебно-тренировочного занятия

Вход: тема обучающего мероприятия.

Выход: сформированное учебно-тренировочное занятие.

Требования:

- тема тренировочного занятия должна совпадать с темой обучающий мероприятий;
- занятие должно реализовываться в условиях организации;
- занятие должно быть направлено на повышение осведомленности сотрудников на реагирование фишинговым угрозам.

Проведение обучения

Вход: сформированный перечень сотрудников на обучение.

Выход: сотрудники осведомленные в части реагирования на угрозы безопасности.

Требования:

- обучение проводится в очно-заочном формате;
- обучение соответствует теме, сформированной на этапе анализа бюллетеней;
- обучение сформировано с учетом понимания процессов целевой аудиторией;
- в обучении участвуют только сотрудники, которым были направлены служебные записки.

Проведение учебно-тренировочного занятия

Вход: сотрудники, прошедшие обучение.

Выход: успешное или неуспешное прохождение тренировочного занятия.

Требования:

- тема тренировочного занятия должна совпадать с темой обучающих мероприятий;
- тренировочное занятие должно быть реализовано в формате фишинговой рассылки;
- занятие должно быть направлено на повышение осведомленности сотрудников на реагирование фишинговым угрозам.

Оценка результатов обучения

Вход: результаты учебно-тренировочного занятия.

Выход: свод по проведенному обучению сотрудников и рекомендации.

Требования:

- результаты должны быть представлены в формате сводной таблицы;
- результаты должны оцениваться согласно метрикам, определенным в данной курсовой работе.

Корректировка

Вход: свод по результатам обучения.

Выход: перечень необходимых корректировок, которые требуется внести в обучение.

Требования:

- должны быть определены сотрудники, которые не прошли учебно-тренировочное занятие;
- определен план корректирующих мероприятий.

Все вышеуказанные функции сведены к общему процессу, отраженному на рис. 1.

При составлении процесса важно учитывать метрики, с помощью которых возможно будет оценить эффективность обучения сотрудников, а также распознать слабые места в обучение, которые требуют корректировки. В процессе обучения сотрудников следует учесть такие метрики как:

1. *Количество уязвимостей, которые возможно закрыть процессом обучения.* Данной метрикой будет оцениваться, количество уязвимостей, в поступившей бюллетени, пригодных для формирования обучения. Если в бюллетени содержится хотя бы одна уязвимость, которую можно закрыть, обучив сотрудников по конкретной теме уязвимости, то весь процесс описанный выше начинает свою работу.

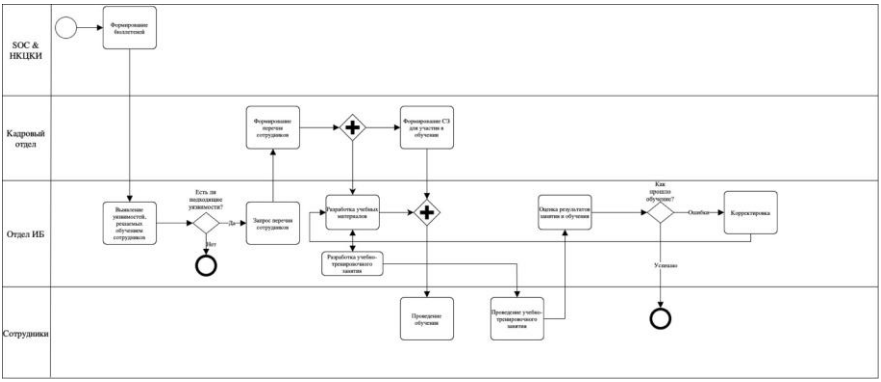


Рис. 1. Схема процесса

2. *Время реагирования сотрудниками на фишинговое письмо, во время проведения учебно-тренировочного занятия.* В процессе прохождения учебно-тренировочного занятия сотрудниками, будет фиксироваться время, затрачиваемое сотрудниками на реагирование. Среднее корректное время реагирование будет определено исходя из сложности направленных фишинговых рассылок, а также предыдущего опыта обучений. Со временем этот показатель примет константное значение.

3. *Пересылка фишингового письма коллегам.* Пользователи могут при получении фишингового письма переслать его коллегам, тем самым увеличив площадь атаки. Данная метрика будет носить бинарный характер и оцениваться, как единица – сообщение было переслано (за исключением пересылки на ящик отдела ИБ), ноль – сообщение не направлялось коллегам.

4. *Открытие ссылки или вложенных файлов, размещенных в фишинговом письме.* При работе с фишинговым письмом будет фиксироваться, осуществлял ли пользователь открытие письма, открытие вложений письма, были ли переходы по ссылкам, оставлялись ли данные на вредоносном ресурсе.

5. *Покрытие целевой группы обучением.* В рамках данной метрики будет оцениваться, сколько сотрудников прошло обучение и сколько из них успешно прошли учебно-тренировочное занятие по реагированию на фишинговую рассылку.

6. *Количество повторных ошибок, после и до обучения сотрудников.* Для сотрудников, кто не с первого раза прошел учебно-тренировочное занятие, будет фиксироваться количество проваленных учебно-тренировочных занятий, проведенных после дополнительного ознакомления с материалами обучения в более расширенном формате.

Алгоритм формирования задания и его ограничения

Для формирования задания предлагается использовать двухуровневую модель, имеющую следующий вид (рис. 2).

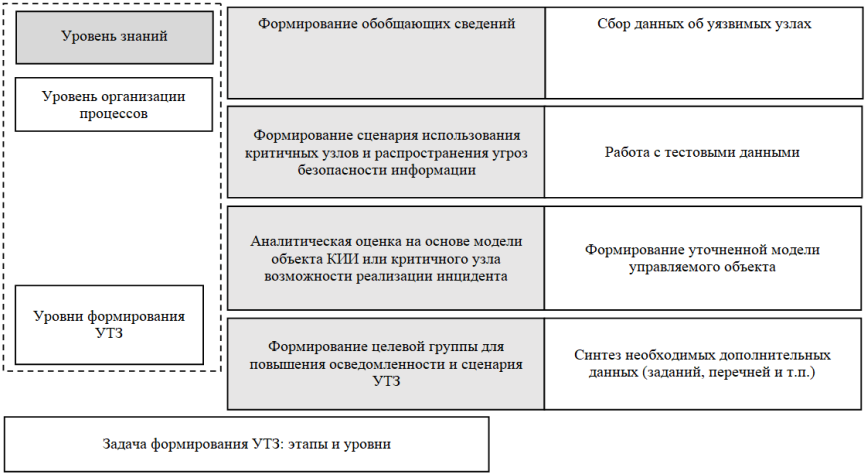


Рис. 2. Формирование учебно-тренировочного задания

Обратим внимание, что для процесса формирования УТЗ потребуется наличие аналитической модели, на основе знаний (сценариев, онтологий) [4–6], позволяющей оценить возможность реализации инцидента. В настоящем исследовании модель строится на основе идеи статьи [7], с использованием гиперграфа как инструмента объединения узлов в случае наличия общих свойств, расширяя представленный подход на случай формирования учебно-тренировочных занятий или киберучений.

В предлагаемой модели для каждого состояния s_i в этом случае имеется гиперграф определенного (статичного либо динамического) вида, где вершинами графа X примем активы (процессы, сценарии, элементы инфраструктуры, в том числе виртуальные), а ребрами графа U – регламентированные или нерегламентированные, скрытые или явные свойства, используемые в принятии решений в задачах детектирования и реагирования на инциденты информационной безопасности, в данном случае в задаче формирования учебно-тренировочного занятия или киберучений. Каждое состояние также имеет определенное для него значение функции оценки допустимости состояния с позиции эксфильтрации, то есть для каждого состояния можно определить, нарушаются ли правила доступа к источникам знаний и/или данных. На уровне источников знаний возможно рассматри-

вать отдельные подразделения, экспертов, экспертные или рабочие группы [7]. Формирование гиперграфа и его использование в исследовании рассмотрено с практической точки зрения, как основа для приведенного ниже алгоритма формирования целевых групп УТЗ.

Далее рассмотрим алгоритм подробнее. Объект КИИ в данном случае представляет собой централизованную автоматизированную систему с единой схемой мониторинга, при этом существует возможность управления уязвимостями на уровне базовых процессов идентификации и анализа.

Для формирования дополнительных данных в рамках этапа синтеза заданий и перечней используются общие выявленные уязвимости (табл. 1). При этом использование общей уязвимости не обязательно должно приводить к компрометации узла (так как в первую очередь рассматриваются превентивные меры), но должен существовать эксплойт и (или) результат тестирования на проникновение, который подтверждает возможность эксплуатации указанной уязвимости злоумышленником.

Далее показан пример пошагового формирования целевой группы для повышения осведомленности и ограничения такого алгоритма (табл. 1).

Алгоритм формирования УТЗ выглядит следующим образом:

1. **Сбор данных об объектах**, включая анализ данных об уязвимостях объектов из общедоступных баз знаний [8]. Этот этап должен предварять формирование модели объекта и в конечном счете создавать базу данных об уязвимых узлах, на которых работает персонал объекта КИИ.

2. **Работа с тестовыми данными**, предполагающая анализ возможности использования уязвимости для реализации техник и тактик, используемых злоумышленником, также на основе открытых данных [8]. Этот этап предполагает тестирование предлагаемых сценариев УТЗ на реальных или виртуальных объектах, в том числе цифровых двойниках объекта КИИ [9].

3. **Формирование модели объекта**, на основе формирования гиперграфа общих свойств (уязвимостей) объектов инфраструктуры, входящих в состав объекта КИИ. На этом этапе необходимо не просто привязаться к возможности реализации угрозы безопасности информации, но и сопоставить реальную возможность возникновения инцидента с бизнес-процессом или технологическим процессом, который будет нарушен в случае его возникновения. Знание процесса, в свою очередь, и позволит перейти к следующему этапу алгоритма.

4**. Формирование целевой группы для повышения осведомленности**, содержащей персонал, задействованный в работе с указанными объектами инфраструктуры.

5. **Формирование сценария учебно-тренировочного задания** (если это возможно), использующего синтезированные данные (в том числе задания, перечни сотрудников, аналитику по угрозам безопасности информа-

ции), в том числе на основе **ретроспективных метрик** по прошлым этапам УТЗ в случае одной и той же целевой группы или завершенным УТЗ для других целевых групп.

В табл. 1 ниже показаны ограничения указанного подхода.

Таблица 1

Алгоритм формирования УТЗ и его ограничения

Пример 1. Централизованная система объекта КИИ			
Объект (пример)	**Действие алгоритма**	**Выходные данные**	**Ограничения**
Защищенный АРМ управления технологическим процессом	Сбор данных об объекте	Базовые угрозы безопасности информации для встроенного ПО Уязвимости встроенного ПО	Требуется высокий уровень внутренней экспертизы или постоянная поддержка внутренней БД уязвимостей и угроз
	Работа с тестовыми данными	Техники и тактики, возможные для использования	Только по известным ретроданным или внешней экспертизе Невозможно оценить приоритетность сценария УТЗ
	Формирование модели объекта	Гиперграф на основе информации о технологическом процессе Граф инфраструктуры объекта	Требуется интерпретация технических данных об объекте на основе информации о технологическом процессе
	Формирование целевой группы для повышения осведомленности	Перечень сотрудников	Невозможно заранее учесть перемещения (увольнения, набор, горизонтальные перемещения) сотрудников
	Формирование сценария УТЗ	Сценарий УТЗ	-

Следовательно, формирование сценария УТЗ на основе предлагаемого алгоритма и изложенного выше процесса возможно автоматизировать, а использование гиперграфа на уровне управления знаниями дает возможность обобщения информации о свойствах узлов (компонентов, подсистем) объекта КИИ.

Этот подход (с учетом показанных выше ограничений) упрощает синтез сценария учебно-тренировочного занятия (киберучений) и может быть применен в практической области.

Далее рассмотрим эксперимент с конструктором сценариев киберполигона Ampire, учитывающий представленные возможности.

Экспериментальная часть

Киберполигон Ampire является учебно-тренировочным комплексом, разработанным для обучения, подготовки и тренировки специалистов в области ИБ по обнаружению, предотвращению и устранению последствий компьютерных атак. Одним из элементов платформы является конфигуратор сценариев, позволяющий создавать тренировки различного уровня сложности, изменять векторы атак, выбирать уязвимые узлы и способы эксплуатации уязвимостей. Далее будет рассмотрен алгоритм формирования УТЗ с использованием платформы.

На базе существующих этапов алгоритма можно выполнить следующее:

Формирование целевой группы для повышения осведомленности, содержащей персонал, задействованный в работе с указанными объектами инфраструктуры:

Используя встроенную базу знаний платформы Ampire, преподаватель проводит анализ актуальных уязвимостей, связанных с инфраструктурой организации. Формируется предварительный перечень компонентов инфраструктуры, требующих включения в учебный сценарий.

На основе выявленных уязвимостей и анализа модели преподаватель формирует целевую группу сотрудников, работающих с выявленными уязвимыми узлами, для повышения осведомлённости и создаёт сценарий учебно-тренировочного занятия с использованием конфигуратора.

На момент написания данной статьи конфигуратор сценариев Ampire предоставляет возможность для выбора между двумя форматами сценария – BlueTeam и CSIRT.

При создании сценария также необходимо выбрать шаблон ИТ-инфраструктуры, моделирующий работу типовой организации, которая включает уязвимые узлы, стандартные для деятельности организации процессы и компьютерные приложения, с которыми работают сотрудники.

Формирование сценария учебно-тренировочного задания (если это возможно), использующего синтезированные данные:

Важной особенностью конфигуратора является возможность детальной настройки параметров атаки. Преподаватель может указать тип нарушителя, выбрать начальную точку компрометации системы и определить конкретный вектор атаки, включая целевой сегмент сети, уязвимый узел, используемую уязвимость и последствия её эксплуатации. После этого сценарий готов для проведения учебно-тренировочного занятия.

Формирование данных для ретроспективного анализа:

После проведения занятия платформа предоставляет отчёт, который включает в себя метрики для оценки эффективности обучения, а именно: среднее время реагирования сотрудников на инцидент, количество закрытых уязвимостей и предотвращенных последствий, а также список сотрудников, который показывает сколько сотрудников прошло обучение и сколько из них успешно его завершили. Эти данные позволяют анализировать текущие результаты и вносить коррективы в программу обучения, что важно для постоянного повышения уровня осведомлённости сотрудников и кибербезопасности в организации.

Выводы

На основе проведенной аналитической работы был разработан детализированный процесс обучения, включающий:

1) определение целей и задач;
2) идентификацию целевой аудитории;
3) создание учебных материалов и использование современных технологий;
4) проведение учебно-тренировочных занятий;
5) мониторинг и совершенствование программы.

Процесс включает четкое определение входов и выходов. Введены метрики для оценки эффективности реализации процесса и внесения корректировок в него.

Применение процесса возможно в практических задачах формирования учебно-тренировочных занятий, а также в оценке информационного риска для автоматизированных систем и объектов КИИ.

Новыми результатами исследования, представленными выше, стали:

1. Подход к формированию задания для проведения киберучений и учебно-тренировочных занятий для персонала объекта КИИ;

2. Алгоритм формирования сценария проведения УТЗ.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Как изменились атаки на российские компании за два года. – URL: https://www.ptsecurity.com/ru-ru/research/analytics/kak-izmenilis-ataki-na-rossiyskie-kompanii-za-dva-goda/#id1 (дата обращения: 03.12.2024).

2. *Левшун Д.С., Гайфулина Д.А., Чечулин А.А., Котенко И.В.* Проблемные вопросы информационной безопасности киберфизических систем // Информатика и автоматизация. – 2020. – Т. 19, № 5. – С. 1050-1088. – DOI 10.15622/ia.2020.19.5.6. – EDN NIWASO.
3. *Зайковский В.Э., Карев А.В., Потапова С.С., Беседина Т.П.* Анализ рисков проекта цифровой трансформации газотранспортного предприятия // Проблемы экономики и управления нефтегазовым комплексом. – 2021. – № 11 (203). – С. 13-16. – DOI 10.33285/1999-6942-2021-11(203)-13-16. – EDN OVNGEE.
4. *Rastogi N. [et al.]*. Malont: An ontology for malware threat intelligence // Deployable Machine Learning for Security Defense. – Cham: Springer International Publishing, 2020. – P. 28-44. – DOI: 10.1007/978-3-030-59621-7_2.
5. *Колесникова Д.С., Верещагина Е.А., Гуляев В.Е.* Построение онтологической модели для предметной области «Информационная безопасность» // Инженерный вестник Дона. –2023. – № 7 (103). – С. 81-90.
6. *Piplai A. [et al.]*. Creating cybersecurity knowledge graphs from malware after action reports // IEEE Access. – 2020. Vol. 8. P. 211691–211703. DOI: 10.1109/ACCESS.2020.3039234.
7. *Золотарев В.В.* Формирование модели управляемого объекта на основе гиперграфа в цикле непрерывного детектирования и реагирования на инциденты информационной безопасности // Прикаспийский журнал: управление и высокие технологии. – 2024. – № 1 (65). – С. 37-44.
8. Матрица MITRE ATT&CK. Positive technologies [Электронный ресурс]. – Режим доступа: https://mitre.ptsecurity.com/ru-RU, свободный (дата обращения: 01.03.2025).
9. *Касимова А.Р., Золотарев В.В., Сафиуллина Л.Х., Балыбердин А.С.* Использование цифрового двойника в задачах управления информационной безопасностью // Прикаспийский журнал: управление и высокие технологии. – 2023. – № 1 (61). – С. 48-58. – DOI 10.54398/20741707_2023_1_48. – EDN GVJYUN.

УДК 004.056

А.А. Краснов

РАЗРАБОТКА СПОСОБА ВИЗУАЛИЗАЦИИ ПРИ РАБОТЕ С ИНЦИДЕНТАМИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Современные системы защиты информации сталкиваются с растущим количеством сложных инцидентов информационной безопасности (ИБ), что требует эффективных методов их анализа и визуализации. Целью данного исследования является практическое применение нового для отрасли метода визуализации инцидентов – визуализация с помощью гиперграфов, и разработка программы, что позволяет выявлять скрытые связи между угрозами, уязвимостями и активами. Задачи исследования включают: анализ возможностей гиперграфов, выявление отличительных черт нового метода и разработку программного инструмента для построения и анализа таких моделей; проведение тестирования программы на реальных сценариях инцидентов. В результате была разработана программа, позволяющая строить гиперграфы, где вершины представляют элементы системы (например, устройства, пользователи), а гиперребра отображают взаимосвязи между ними (например, угрозы, общие свойства). Программа поддерживает функции добавления, удаления и анализа сцен, а также выполнение предварительного анализа для выявления уязвимых элементов. Ключевым преимуществом является возможность наглядного представления сложных зависимостей в системе, что облегчает принятие решений при ликвидации инцидентов. Разработанный инструмент может быть использован специалистами по ИБ для анализа и прогнозирования угроз.

Ключевые слова: *информационная безопасность, визуализация данных, гиперграфы, инциденты ИБ, анализ угроз.*

Modern information security (IS) systems face a growing number of complex security incidents, necessitating effective methods for their analysis and visualization. The aim of this research is the practical application of a new method in the field—visualization using hypergraphs—and the development of a program that allows for the identification of hidden connections between threats, vulnerabilities, and assets. The research tasks include: analyzing the capabilities of hypergraphs, identifying the distinctive features of the new method, and developing a software tool for constructing and analyzing such models; testing the program on real-world incident scenarios. As a result, a program was developed that allows the construction of hypergraphs, where nodes represent system elements (e.g., devices, users), and hyperedges depict relationships between them (e.g., threats, shared properties). The program supports functions for adding, removing, and analyzing scenes, as well as performing preliminary analysis to identify vulnerable elements. A key advantage is the ability to visually represent complex dependencies within a system, facilitating decision-making during incident response. The developed tool can be used by IS professionals for threat analysis and prediction.

Keywords: *information security, data visualization, hypergraphs, IS incidents, threat analysis.*

В начале требуется уделить время на раскрытие понятия, о котором дальше пойдет речь. Гиперграф представляет собой расширенную графовую модель, где связи (гиперребра) могут объединять произвольное количество узлов. По сути, это обобщение классического графа, в котором ребра способны соединять не только две вершины, но и любые подмножества множества вершин. Такая структура активно применяется в задачах моделирования сложных систем, включая электрические цепи, где требуется отразить взаимодействие между несколькими компонентами одновременно [1].

Отличительной чертой гиперграфов является их уникальная способность представлять многомерные связи между элементами системы, что особенно ценно при изучении сложных сценариев развития атак в сфере информационной безопасности. В современном мире киберугрозы становятся все более изощренными, и для их анализа требуются инструменты, способные отображать не только линейные взаимодействия, но и сложные многоуровневые зависимости. Именно здесь гиперграфы демонстрируют свое ключевое преимущество перед традиционными графами [2].

Традиционные графы ограничены тем, что каждое ребро может соединять только две вершины, что значительно сужает возможности моделирования реальных систем. В отличие от них, гиперграфы позволяют строить связи между произвольным количеством узлов через одно гиперребро. Это дает возможность наглядно иллюстрировать множественные взаимодействия объектов, что существенно упрощает задачу визуального анализа и интерпретации данных. Например, одна атака может охватывать несколько узлов сети или сервисов, что легко отражается через одно гиперребро. Такой подход позволяет не только выявить последовательность шаговзлоумышленника отначальной точки проникновения к конечной цели (вертикальный путь атаки), но и учесть горизонтальные воздействия, затрагивающие широкий круг элементов системы.

Этот подход способен существенно повысить эффективность анализа, так как позволяет фокусироваться на основных свойствах узлов, которые стали мишенью для атаки, и оперативно выявлять все связанные элементы системы с подобными свойствами. Например, при обнаружении уязвимости в одном из узлов специалисты смогут мгновенно определить весь массив узлов, которые могут быть подвержены аналогичной угрозе.

Такой подход особенно эффективен при расследовании инцидентов, связанных с распространением вредоносного программного обеспечения (ВПО), когда необходимо быстро выявить все зараженные устройства или уязвимые участки сети. Эта проблема описывается в рамках матрицы Mitre ATT&CK и относится к тактике, известной как «Боковое перемещение» (или «Перемещение внутри периметра» в переводе от Positive Technologies). Боковое перемещение включает в себя методы, которые злоумышленники применяют для проникновения в удаленные системы сети и управления ими [3, 4].

Чтобы достичь своей конечной цели, атакующие часто исследуют сеть, чтобы найти нужные ресурсы и получить к ним доступ. Для этого они могут использовать множество систем и учетных записей. Злоумышленники могут задействовать собственные инструменты удаленного доступа для осуществления бокового перемещения либо воспользоваться легальными учетными данными, эксплуатируя встроенные сетевые средства и инструменты операционной системы. Такая тактика позволяет им действовать скрытно в рамках одной сети и минимизировать вероятность обнаружения.

Кроме того, применение гиперграфов дает возможность создавать модели, которые учитывают сложные взаимодействия между различными элементами информационных систем, такими как серверы, рабочие станции, маршрутизаторы или программные приложения. Это позволяет более детально анализировать структуру атак и выявлять уязвимые места в системе защиты, что, в свою очередь, способствует разработке эффективных мер противодействия киберугрозам. Помимо значительного упрощения процесса расследования инцидентов, данный подход также облегчает подготовку отчетной документации и проведение презентаций для специалистов и руководства, поскольку предоставляет четкое и структурированное представление о ситуации.

Для демонстрации практического применения гиперграфов в расследовании инцидентов информационной безопасности была разработана последовательность действий, переводящая абстрактные концепции в конкретные аналитические выводы. Гиперграфы выступают не только как формализм для описания сложных взаимодействий в инфраструктуре, но и как мощный инструмент для системного анализа инцидентов ИБ. Рассмотрение реального сценария на основе моделирования позволило выделить ключевые преимущества их использования:

1. Визуализация векторов атаки: гиперграфы позволяют наглядно представить все возможные векторы атаки, включая прямые и косвенные зависимости между элементами системы. Это помогает специалистам по информационной безопасности быстро определить потенциальные точки проникновения, закрепления и попыток боковых переходов злоумышленников.

2. Выявление критических узлов: с помощью гиперграфов можно идентифицировать узлы, компрометация которых может привести к большим убыткам компании или просто стать удобным плацдармом для последующих злонамеренных действий злоумышленников. Такие узлы становятся приоритетными объектами защиты, что позволяет сосредоточить усилия на их укреплении.

3. Прогнозирование развития инцидента: анализ связей в гиперграфе дает возможность спрогнозировать дальнейшее развитие атаки. Это позволяет оценить риски, связанные с каждым этапом инцидента, и подготовить эффективные контрмеры для минимизации ущерба.

Для наглядности был смоделирован пример инцидента, демонстрирующий распространение ВПО в корпоративной сети. Сценарий выглядит следующим образом: компания использует сегментированную сеть, состоящую из трех подсетей, каждая из которых включает три автоматизированных рабочих места (АРМ). Подсети соединены маршрутизатором, а также имеется общий файловый сервер, доступный для АРМ из первой и второй подсети. Инцидент начался с того, что АРМ1 из первой подсети (.10) был заражен вирусом через фишинговую ссылку, отправленную на почту сотрудника. На входе в сеть установлен межсетевой экран (МЭ).

На основе этого сценария были выделены ключевые группы и элементы системы, которые затем были записаны в файл. Далее этот файл был импортирован в разработанную программу для визуализации инцидентов с использованием гиперграфов. Программа позволила построить гиперграф, отображающий все взаимосвязи между элементами сети, включая подсети, АРМ, файловый сервер и маршрутизатор. Полученный гиперграф представлен на рис. 1.

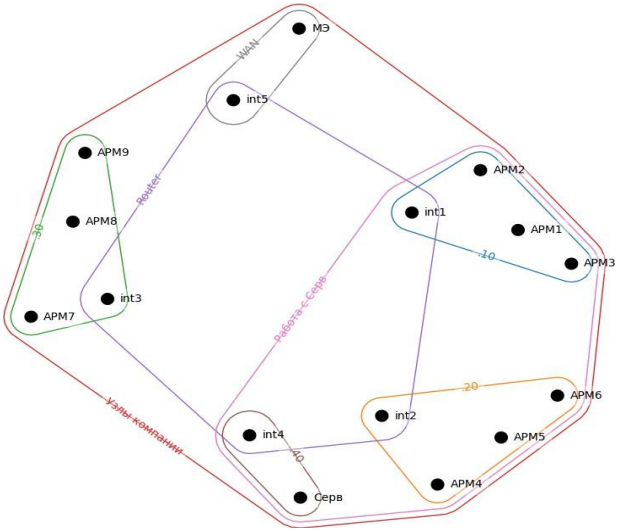


Рис. 1. Пример использования гиперграфа

На гиперграфе отчетливо видны точки коммуникации, включая интерфейсы маршрутизатора, отдельные подсети и возможные взаимодействия между ними. Например, подсети .10 и .20 имеют доступ к общему серверу, что создает потенциальные пути для распространения сетевого вредоносного программного обеспечения (ВПО). Исходя из этого можно

предположить следующий сценарий: ВПО сначала поражает подсеть .10, затем через взаимодействие с сервером заражает его, а далее распространяется на подсеть .20.

Учитывая, что взаимодействие между подсетями .10 и .40, а также .20 и .40 осуществляется через маршрутизацию, маршрут распространения ВПО может быть описан как .10 -> .40 -> .20.

Благодаря динамической настройке отображаемых свойств гиперграфа, его можно легко масштабировать, добавляя новые данные или убирая ненужные. Например, после анализа стало ясно, куда может распространиться ВПО, а куда нет. Информация о тех узлах, которые не находятся под угрозой, была исключена из визуализации, так как она является избыточной. После этого все узлы, находящиеся под угрозой, были объединены в одну группу на основе выявленного признака. Также было установлено, что источником ВПО является сеть Интернет.

В процессе расследования было выявлено, что вредоносное программное обеспечение (ВПО) распространяется через определенный порт, который на большинстве АРМ заблокирован. Это существенно сужает область возможной атаки, так как ВПО не может использовать этот путь для распространения на такие АРМ. Учитывая эту информацию, гиперграф был обновлен, чтобы отразить новые данные.

Кроме того, было добавлено новое свойство – "наличие критически важной информации" (КритИнф), которое указывает, на каких АРМ хранятся чувствительные данные, представляющие особую ценность для организации. Это позволяет выделить приоритетные узлы, требующие усиленной защиты.

Лишние элементы, которые не имеют отношения к текущему сценарию атаки или не влияют на его развитие, были исключены из визуализации. Такой подход помогает сосредоточиться на ключевых аспектах инцидента и минимизировать информационный шум. Обновленный гиперграф, учитывающий эти изменения, представлен на рис. 2.

С помощью последней итерации гиперграфа можно легко выделить ключевые зоны риска в сети. Например, становится очевидно, что АРМ 5 находится в относительной безопасности благодаря контролю портов, АРМ 7 потенциально вообще не подвержен угрозе, а АРМ 2, напротив, оказывается в непосредственной зоне риска. Такая визуализация позволяет своевременно расставить приоритеты при реагировании на инцидент, что может предотвратить утечку чувствительных данных и минимизировать возможный ущерб.

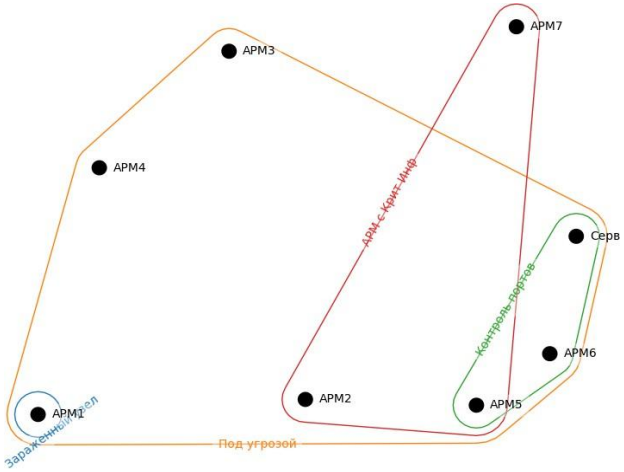


Рис. 2. Развитие расследования инцидента

Однако использование гиперграфов имеет свои особенности, которые могут быть рассмотрены как ограничения:

1. Невозможность учесть человеческий фактор. Гиперграфы представляют собой абстрактные математические модели, которые не способны отразить влияние человеческих действий. Ошибки оператора, злонамеренное поведение сотрудников или успешные атаки методом социальной инженерии могут существенно изменить ход распространения вредоносного программного обеспечения (ВПО), приводя к неожиданным последствиям. Эти факторы остаются за рамками анализа, что требует дополнительного внимания со стороны аналитиков.

2. Субъективность выбора свойств для построения гиперграфа. Процесс создания гиперграфа зависит от опыта и знаний аналитика, который самостоятельно определяет значимые элементы сети и связи между ними. Если список активных свойств окажется избыточным, это может снизить читабельность графа. В то же время, неправильный выбор свойств может привести к созданию неполной или неточной модели, затрудняя анализ и принятие решений.

Для улучшения процесса анализа и снижения влияния субъективных факторов целесообразно разработать методику, которая будет помогать аналитикам в выборе свойств для построения гиперграфов. Такая методика должна учитывать специфику инфраструктуры конкретной организации, а также цели и задачи анализа. Однако создание универсальной методики пред-

ставляется сложной задачей, поскольку каждая организация уникальна и имеет свои особенности в области безопасности. Кроме того, инциденты могут существенно различаться, требуя индивидуального подхода к их анализу.

Важно отметить, что разработка такой методики должна осуществляться совместно с экспертами, глубоко понимающими инфраструктуру организации. Только такой подход позволит обеспечить адекватный выбор свойств для построения гиперграфов и минимизировать риск ошибок. Это особенно важно, поскольку качество анализа напрямую зависит от точности и полноты модели, представленной гиперграфом.

Дальше речь пойдет непосредственно о разработанном решении. Программа была создана на языке Python с использованием библиотеки HyperNetX, которая предназначена для работы с гиперграфами и их визуализации. На рис. 3 представлен интерфейс программы, демонстрирующий её основные функциональные элементы [5].

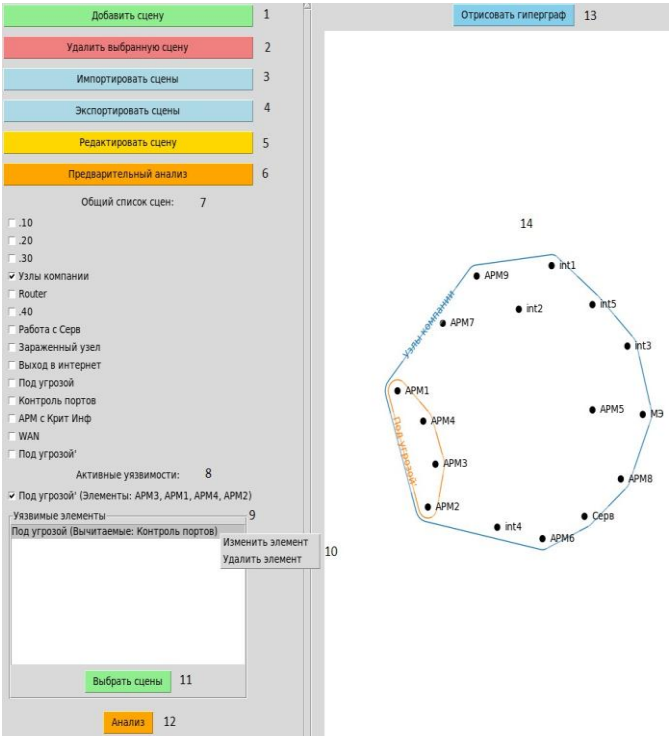


Рис. 3. Интерфейс программы

Далее будут рассмотрены основные элементы и функции программы в соответствии с нумерацией на рис. 3:

1. Добавить сцену – кнопка, которая позволяет добавить новую сцену и ее элементы, перечисляя их через пробел.

2. Удалить выбранную сцену – кнопка, позволяющая удалить сцену из общего списка сцен.

3. Импортировать сцены – кнопка, которая позволяет добавить сцены записанные в файле расширения json, где они должны быть формализованы под стандарт работы библиотеки HyperNetX имея следующую структуру:

```
{
"сцена 1": ["элемент 1", "элемент 2"…],
"сцена 2": ["элемент 2", "элемент 3"…],
…
…
}
```

4. Экспортировать сцены – кнопка, отвечающая за сохранения текущих сцен в файл с расширением json.

5. Редактировать сцену – кнопка, позволяющая изменить название сцены или хранящиеся в ней элементы.

6. Предварительный анализ – кнопка, которая необходима для демонстрации или сокрытия элементов 9-12, которые в свою очередь представляют отдельный блок «анализа», хоть анализ тут скорее условный и состоит в том, чтоб из одной сцены удалить элементы, которые есть в другой сцене. Это необходимо для облегчения работы, так, например, можно автоматически из сцены «Под угрозой» исключить элементы сцены «Контроль портов», получая чистый список «Под угрозой'».

7. Общий список сцен – список сцен из которого можно выбрать интересующие свойства системы, по которым после будет отрисован гиперграф.

8. Активные уязвимости – список сцен, прошедших стадию анализа, из которых исключены элементы других сцен.

9. Уязвимые элементы – интерактивный список сцен, которые добавляются из «Общего списка сцен» с помощью кнопки 11.

10. Контекстное меню – список действий, которые можно совершить с элементом списка «Уязвимые элементы», при нажатии на «Изменить элемент» появляется диалоговое окно для выбора сцен из «Общего списка сцен», за исключением тех, что уже добавлены в список «Уязвимые элемен-

ты». Элементы выбранных сцен будут исключаться из сцены, выбранной при взаимодействии с кнопкой 11. Так же есть пункт «Удалить элемент», который удаляет «первичную сцену» и привязанные к ней сцены с взаимоисключающими элементами.

11. Выбрать сцены – кнопка, добавляющая сцены в список «Уязвимые элементы», это выбор первоначальных сцен, от которых будут отниматься элементы сцен, описанных в пункте 10.

12. Анализ – кнопка запускающая взаимоисключение элементов из сцен, выбранных при формировании списка «Уязвимые элементы» и добавляет результат в список «Активные угрозы».

13. Отрисовать гиперграф – кнопка выводящая гиперграф на основе выбранных элементов из «Общего списка сцен» и «Активных угроз».

14. Гиперграф – зона отрисовки гиперграфа.

Конечная цель разработки данного подхода заключается в улучшении эффективности реагирования на инциденты путем адаптации предложенных методов и инструментов для использования в повседневных операциях центров мониторинга и реагирования на инциденты информационной безопасности (SOC – Security Operation Center). Применение платформ автоматизации и оркестрации безопасности (SOAR – Security Orchestration, Automation and Response) позволит интегрировать полученные результаты в существующие рабочие процессы, что повысит скорость и точность принятия решений, а также обеспечит согласованность действий при ликвидации последствий инцидентов. Уже на данном этапе возможно взаимодействие с подобными системами, необходим только коннектор для формализации данных из SOAR, что не является большой проблемой.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Антонова, Балакин, Гречишкина, Клыгин, Кузнецов*. Информационные процессы. – 2021. – Т. 21, № 4. – С. 203-210.
2. *Золотарев В.В.* Формирование модели управляемого объекта на основе гиперграфа в цикле непрерывного детектирования и реагирования на инциденты информационной безопасности // Прикаспийский журнал: управление и высокие технологии. – 2024. – № 1 (65). – С. 37-44.
3. Матрица MITRE ATT&CK. Positive technologies. [Электронный ресурс]. – Режим доступа: https://mitre.ptsecurity.com/ru-RU, свободный (дата обращения: 01.03.2025).
4. Lateral Movement, MITRE ATT&CK. [Электронный ресурс]. – Режим доступа: https://attack.mitre.org/tactics/TA0008/, свободный (дата обращения: 01.03.2025).
5. Официальный сайт библиотеки HyperNetX (HNX) [Электронный ресурс]. – Режим доступа: https://hypernetx.readthedocs.io/en/latest/index.html, свободный (дата обращения: 23.01.2025).

УДК 004.056

А.М. Кукарцев

Сибирский государственный университет науки и технологий
им. акад. М.Ф. Решетнёва, Россия, г. Красноярск

О РЕАЛИЗАЦИИ «СТРИБОГ» ДЛЯ РЕСУРСОЭФФЕКТИВНЫХ ПЛАТФОРМ

В работе рассматривается возможность промышленных реализаций российского алгоритма контрольного суммирования ГОСТ Р 34.11-2012 / ГОСТ 34.11-2018 для ресурсоэффективных платформ, используемых в устройствах IoT и IIoT. Анализируются принципиальные вопросы необходимости и законности использования реализаций в правовом поле России. Предлагается модификация алгоритма на основе предвычислений. Она позволяет получить в реализациях характеристики промышленного уровня. Описывается математический аппарат, доказывающий эквивалентность алгоритмов модифицированного и оригинального. Представляется готовое решение промышленного уровня для платформ ATmega328P и AMD64, являющееся свободным программным обеспечением под лицензией Mozilla Public License Version 2.0. Приводятся результаты испытаний предлагаемых решений и оценка их эффективности.

Ключевые слова: *контрольное суммирование, HMAC, ГОСТ Р 34.11-2012, ГОСТ 34.11-2018, ATmega328P, оптимизация, предвычисления, IoT, IIoT, промышленное программирование, язык C++.*

The article discusses the possibility of industrial implementations of the Russian checksum algorithm GOST R 34.11-2012 / GOST 34.11-2018 (Request for Comments 6986 of Internet Engineering Task Force) for resource-efficient platforms. It is characterized by low consumer characteristics. These characteristics are performance (tens of megahertz) and RAM (units of kibibytes). This leads to a significant reduction in the cost and energy consumption of the final information technology product based on resource-efficient platforms. For this reason, such products are used in IoT and IIoT. They act as data collection devices (temperature, humidity, electricity, chemical sensors, radiation, etc.), executable devices (remote power management devices, lighting control, unmanned aerial vehicles, etc.) and control devices (e.g. operator control terminals). There are complexities in developing implementations of the Russian checksum algorithm in IoT and IIoT devices. They include issues of legal support for both the development process and the operation process. They include achieving industrial-grade consumer characteristics sufficient to ensure the competitiveness of finished IoT and IIoT devices. To solve the issues of legal support, the work analyzes the fundamental necessity and legality of using the implementations of the algorithm in question in the legal field of Russia. To become consumer characteristics of the industrial level, a modification of the algorithm based on pre-calculations is proposed. The mathematical apparatus is described. It proves the equivalence of the modi-

fied and original algorithms. A ready-made industrial-grade solution for the ATmega328P and AMD64 platforms is presented. The solution is implemented in accordance with modern requirements for industrial software product development. It is free software licensed under Mozilla Public License Version 2.0. Test results are provided for the provided solution. In conclusion, the effectiveness of the solutions obtained is assessed, and future tasks for the development of the work are formulated. The solutions obtained are available in the git repository https://gitflic.ru/project/penta/isrdt-stribog/. It is part of the free project "information systems research and development tools", isrndt.

Keywords: *checksum, HMAC, GOST R 34.11-2012, GOST 34.11-2018, RFC 6986, ATmega328P, optimization, pre-calculation, IoT, IIoT, industrial programming, C++ language.*

Целеполагание

Контроль целостности является одной из основных функций подавляющего большинства систем обеспечения информационной безопасности. Реализации российского алгоритма контрольного суммирования (далее – КС), описанные в ГОСТ Р 34.11-2012 / ГОСТ 34.11-2018 [1] (далее – защитное преобразование «Стрибог», ЗП «Стрибог»), а также в Рабочем предложении № 6986 *(RFC* 6986) Инженерного совета Интернета (*Internet Engineering Task Force, IETF*) [2] применяются для расчёта КС при контроле целостности такими методами как КС (т.н. хеширование), имитозащита (т.н. *hash-based message authentication code* или *HMAC*) и электронная подпись.

Современные информационные системы усложняются, в том числе, в сторону использования ресурсоэффективных платформ. Составляющие таких информационных систем представляют собой самостоятельные вычислительные узлы, выполняющие конкретные задачи. Такими узлами могут выступать: устройства сбора сведений (датчики температуры, влажности, электроэнергии, химические датчики, радиации и пр.), исполнимые устройства (устройства удалённого управления питанием, управления освещением, беспилотные летательные аппараты и пр.) и управляющие устройства (например терминалы управления оператора). В совокупности они образуют т. н. Интернет вещей (*Internet of Things*, далее – *IoT*) и т.н. промышленный Интернет вещей (*Industrial Internet of Things*, далее – *IIoT*). Они используются не только в системах «умный дом» в гражданском секторе, но и в автоматизированных системах управления технологическими процессами в промышленности, и в военном секторе в том числе.

Отличительной особенностью этих устройств является их ресурсоэффективность. Она фактически определяется выполнением задач устройством минимально необходимыми ресурсами. Она является фундаментальным требованием, потому что определяет два основных экономических параметра: цену конечной продукции и её низкое энергопотребление. Это позволило построить *IoT*, число устройств которого в 2017 году превысило население планеты [3].

Требование ресурсоэффективности делает задачу реализации ЗП «Стрибог» нетривиальной. В качестве примера макетной ресурсоэффективной платформы можно рассмотреть *Arduino UNO* на базе микроконтроллера *ATMega*328*P*. Её основные характеристики: 16 МГц частота, 32 КиБ микропрограмма и 2 КиБ оперативной памяти, разрядность 8 бит [4]. Согласно [1] только массив данных российского алгоритма КС составляет 1600 байт. Это число не учитывает переменные алгоритма (не менее 384 байта). В результате само ЗП «Стрибог» не получится разместить в такой платформе, не говоря уже о реализациях целевых алгоритмов, которые должно выполнять устройство.

Цель: исследовать возможность реализуемости российского алгоритма КС в ресурсоэффективных платформах и предложить такую реализацию.

Работа выполнена во исполнение:

- п.п. з) п. 23 и п.п. а) и б) п. 25 части IV Указа Президента России № 646 от 05.12.2016 [5] **в направлении достижения стратегических целей обеспечения информационной безопасности России в области государственной и общественной безопасности, и в направлении достижения стратегических целей обеспечения информационной безопасности России в экономической сфере**;
- п. 55, п. 56 и п.п. 12 и п.п. 13 п. 57 части IV Указа Президента России № 400 от 02.07.2021 [6] **как часть государственной политики направленной на достижение цели обеспечения информационной безопасности России**.

Откуда можно заключить, что массовое применение (не только в части официальных требований) ЗП «Стрибог», взамен западных аналогов, в программных, аппаратных и программно-аппаратных средствах и системах, реализующих информационные технологии, является обязательным и приоритетным направлением развития отрасли информационных технологий в России.

Работа направлена на устранение следующих причин слабого использования ЗП «Стрибог» среди разработчиков информационных систем в России:

1. Иррациональный необоснованный страх необходимости лицензирования деятельности при разработке, производстве, реализации и эксплуатации информационных систем на базе ЗП «Стрибог», а так же иррациональный необоснованный страх необходимости сертификации разрабатываемого продукта.

2. Высокий порог вхождения при реализации ЗП «Стрибог». Требуются специализированные узкие знания области дискретной математики, криптографии и промышленного программирования для разработки реализации ЗП «Стрибог» промышленного уровня.

3. Необходимость проведения научных исследований с целью выведения потребительских характеристик (потребляемые ресурсы и производительность) реализаций на требуемый промышленный уровень, такой, что использование реализации в ресурсоэффективных платформах рыночно жизнеспособно.

4. Низкое число готовых реализаций промышленного уровня в репозиториях свободного программного обеспечения. В результате, или готовые разработки (т.н. «студенческие» разработки) не эффективны по производительности, или разработки промышленного уровня, но не являются свободными. Например промышленное проприетарное решение `libgost` находится в компоненте `non-free` в официальном репозитории *Astra Linux Special Edition* (по адресу https://download.astralinux.ru/astra/stable/1.7_x86-64/repository-base/).

Предлагаемые решения и их испытания

В работе проведены исследования по оптимизации ЗП «Стрибог». Разработаны модификации алгоритма [1] на основе предвычислений. **Математически доказаны** эквивалентность этих решений оригинальному алгоритму [1].

В труде [7] авторы рассматривают реализации ЗП «Стрибог» для увеличения производительности через похожий аппарат предвычислений, но не приводят доказательства эквивалентности модернизированного ими алгоритма и оригинального. Они ориентируются на реализацию ЗП «Стрибог» для *NVIDIA GPU*. В труде [8] автор рассматривает реализации ЗП «Стрибог» на различных высокопроизводительных платформах, но не для ресурсоэффективных.

В работе созданы промышленные реализации ЗП «Стрибог» для *ATmega*328*P* и *AMD*64. Они размещены на Первой Российской Платформе для разработчиков *GitFlic* по адресу https://gitflic.ru/project/penta/isrdt-stribog/.

Основная кодовая база является свободным программным обеспечением, и останется свободным, предоставляемым по лицензии *Mozilla Public License Version* 2.0 с полным текстом которой можно ознакомиться по адресу https://mozilla.org/MPL/2.0/. Кодовая база системы тестирования является свободным программным обеспечением, предоставляемым по лицензии *The MIT License (MIT)* с полным текстом которой можно ознакомиться по адресу https://mit-license.org/. В отличие от *Global Public License, MPL* 2.0 не требует раскрытия исходных кодов всего проекта, а только лишь изменённой части предлагаемых решений «*MPL* заражает файл, а не проект». Поэтому их можно интегрировать как в свободные, так и в проприетарные проекты.

Реализации написаны на базе `stdint.h` на языке *C++* без использования специализированных библиотек и виртуализации классов. Поэтому могут свободно переноситься между различными компиляторами *C++*. Раз-

работка произведена в операционной систем на базе ядра *Linux*. Они соответствуют основным требованиям к разработке промышленного программного продукта, а именно использования: системы контроля версий, системы модульного тестирования, анализа покрытия кода тестами, профилирования, документирования и распространения с помощью пакетных систем.

Испытания для *ATmega*328*P* (табл. 1) проводились на *Arduino UNO*.

Таблица 1

Результаты испытаний решения для *ATmega*328*P*

Ветка репозитория	Версия	Тест 512 байт, мксек	Объём микропрограммы, байт	Объём ОЗУ, байт	Примечание
ATmega328P	0.0.0.0.0	526 184	3 732 (11%)	402 (19%)	Оригинальный
ATmega328P	0.2.4.0.0	214 196	5 148 (15%)	396 (19%)	Предвычисления 4 бита
ATmega328P	0.2.5.0.0	137 608	19 138 (59%)	396 (19%)	Предвычисления 8 бит

Испытания для *amd*64 (табл. 2) проведены на стенде с параметрами: процессор *Intel*(*R*) *Core*(*TM*) *i*7-7820*X CPU* @ 3.60*GHz*, 2 ядра; оперативная память 8 ГиБ; подкачка отключена; операционная система *Astra Linux Special Edition* 1.7.5.9; компилятор *gcc version* 8.3.0 (*AstraLinuxSE* 8.3.0-6).

Таблица 2

Результаты испытаний решения для *amd*64

Версия	Тест 1 КиБ, мксек	Тест 1 МиБ, мксек	Тест 100 МиБ, мксек	Тест 500 МиБ, мксек	Объём констант, байт	Примечание
0.0.0.0.0	1 527	1 221 112	–	–	1 600	Оригинальный алгоритм
0.0.0.0.1	1 014	843 183	–	–		
0.0.0.0.2	964	790 697	–	–		
0.0.0.0.3	721	609 202	–	–		
0.1.0.0.0	49	41 402	3 919 981	19 647 150	16 384	Предвычисления 8 бит
0.1.0.0.1	10	8 973	866 879	4 365 497		
0.1.0.0.2	**10**	**8 392**	**820 354**	**4 096 603**		
0.1.0.0.3	11	9 169	891 547	4 449 730		

В версиях 1.0.0.0.X добавлено «быстрое» вычисление единственного блока 512 байт для анализа безопасности парольных систем. Максимальная скорость «быстрых» вычислений единственного блока 512 байт в версии 1.0.0.0.2 составила 20 млн. единственных блоков за 3 003 002 микросекунд.

Опираясь на табл. 1 и 2 можно оценить прирост производительности за счёт разработанных математических методов независимо от параметров испытательного стенда (замеры произведены в равных условиях):

- *ATmega328P:* версия 0.2.4.0.0: в **2,67 раз**; версия 0.2.5.0.0: в **3,82** раза.
- *AMD64*: версия 0.1.0.0.2: в **90,55 раз**.

Предложенные решения являются частью проекта «Инструменты исследований и разработок информационных систем» (*information systems research and development tools, isrdt*). Проект является полностью свободным и будет оставаться свободным. Согласно сорокалетним традициям поддержки свободного программного обеспечения приведём ссылки, по которым можно присоединиться к проекту и/или поддержать проект в любой форме.

Проект `isrdt-stribog`
https://gitflic.ru/project/penta/isrdt-stribog/

Telegram Подписаться на бесплатный телеграмм-канал (тут можно подписаться на частный телеграмм-канал с обсуждением)

Проект «Инструменты исследований и разработок информационных систем» (information systems research and development tools, isrdt) в части исследований и разработок реализаций ГОСТ Р 34.11-2012 / ГОСТ 34.11-2018 isrdt-stribog выполнен при поддержке программы стратегического академического лидерства «Приоритет-2030» СибГУ им. М.Ф. Решетнёва, г. Красноярск.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ГОСТ 34.11–2018 Межгосударственный стандарт. Информационная технология. Криптографическая защита информации. Функция хеширования. Издание официальное. – М.: Стандартинформ, 2018. – 23 с.
2. RFC 6986 GOST R 34.11-2012: Hash Function [Электронный ресурс]. – URL: https://datatracker.ietf.org/doc/html/rfc6986 (дата обращения: 24.02.2025).
3. Liam Tung IoT devices will outnumber the world's population this year for the first time [Электронный ресурс]. – URL: https://www.zdnet.com/article/iot-devices-will-outnumber-the-worlds-population-this-year-for-the-first-time/#ftag=RSSbaffb68 (дата обращения: 24.02.2025).
4. Arduino UNO R3. Product Reference Manual. SKU: A000066 [Электронный ресурс]. – URL: https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf (дата обращения: 24.02.2025).
5. Указа Президента России № 646 от 05.12.2016 «Об утверждении Доктрины информационной безопасности Российской Федерации» [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс».
6. Указ Президента России № 400 от 02.07.2021 «О стратегии национальной безопасности Российской Федерации» [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс».
7. *Kazymyrov O., Shevchuk O.* Implementation of hash function Stribog. [Электронный ресурс]. – URL: https://github.com/okazymyrov/stribog (дата обращения 24.02.2025).
8. *Lebedev P.A.* Comparison of old and new cryptographic hash function standards of the Russian Federation on CPUs and NVIDIA GPUs // Матем. вопр. криптогр. – 2013. – № 4:2. – P. 73-80.

УДК 004.056

А.М. Кукарцев

Сибирский государственный университет науки и технологий
им. акад. М.Ф. Решетнёва Россия, г. Красноярск

О РЕАЛИЗАЦИИ «КУЗНЕЧИК» ДЛЯ РЕСУРСОЭФФЕКТИВНЫХ ПЛАТФОРМ

В работе рассматривается возможность промышленных реализаций российского алгоритма симметричного шифрования «Кузнечик» в режиме гаммирования с обратной связью по выходу согласно ГОСТ Р 34.12-2015 / ГОСТ 34.12-2018 и ГОСТ Р 34.13-2015 / ГОСТ 34.13-2018 для ресурсоэффективных платформ, используемых в устройствах IoT и IIoT. Анализируются принципиальные вопросы необходимости и законности использования реализаций в правовом поле России. Предлагается модификация алгоритма на основе предвычислений. Она позволяет получить в реализациях характеристики промышленного уровня. Описывается математический аппарат, доказывающий эквивалентность алгоритмов модифицированного и оригинального. Представляется готовое решение промышленного уровня для платформ ATmega328P и AMD64, являющееся свободным программным обеспечением под лицензией Mozilla Public License Version 2.0. Приводятся результаты испытаний предлагаемых решений и оценка их эффективности.

Ключевые слова: *Кузнечик, OFB, ГОСТ Р 34.12-2015, ГОСТ 34.12-2018, ГОСТ Р 34.13-2015, ГОСТ 34.13-2018, ATmega328P, оптимизация, предвычисления, IoT, IIoT, промышленное программирование, язык C++.*

The article discusses the possibility of industrial implementations of the Russian symmetric encryption algorithm "Kuznechik" GOST R 34.12-2015 / GOST 34.12-2018 and GOST R 34.13-2015 / GOST 34.13-2018 (Request for Comments 7801 of Internet Engineering Task Force) in the gamma mode with output feedback for resource-efficient platforms. It is characterized by low consumer characteristics. These characteristics are performance (tens of megahertz) and RAM (units of kibibytes). This leads to a significant reduction in the cost and energy consumption of the final information technology product based on resource-efficient platforms. For this reason, such products are used in IoT and IIoT. They act as data collection devices (temperature, humidity, electricity, chemical sensors, radiation, etc.), executable devices (remote power management devices, lighting control, unmanned aerial vehicles, etc.) and control devices (e.g. operator control terminals). There are complexities in developing implementations of the Russian symmetry cryptographic algorithm "Kuznechik" in IoT and IIoT devices. They include issues of legal support for both the development process and the operation process. They include achieving industrial-grade consumer characteristics sufficient to ensure the competitiveness of finished IoT and IIoT devices. To solve the issues of legal support, the work analyzes the fundamental necessity and legality of using the implementations of the algorithm in ques-

tion in the legal field of Russia. To become consumer characteristics of the industrial level, a modification of the algorithm based on pre-calculations is proposed. The mathematical apparatus is described. It proves the equivalence of the modified and original algorithms. A ready-made industrial-grade solution for the ATmega328P and AMD64 platforms is presented. The solution is implemented in accordance with modern requirements for industrial software product development. It is free software licensed under Mozilla Public License Version 2.0. Test results are provided for the provided solution. In conclusion, the effectiveness of the solutions obtained is assessed, and future tasks for the development of the work are formulated. The solutions obtained are available in the git repository https://gitflic.ru/project/penta/isrdt-kuznechik/. It is part of the free project "information systems research and development tools", isrndt.

Keywords: *Kuznechik, OFB, GOST R 34.12-2015, GOST 34.12-2018, GOST R 34.13-2015, GOST 34.13-2018, RFC 7801, ATmega328P, optimization, pre-calculation, IoT, IIoT, industrial programming, C++ language.*

Целеполагание

Симметричное шифрование является одной из основных функций подавляющего большинства систем обеспечения информационной безопасности. Реализации российского алгоритма симметричного шифрования «Кузнечик», описанные в ГОСТ Р 34.12-2015 / ГОСТ 34.12-2018 [1] (далее защитное преобразование «Кузнечик», ЗП «Кузнечик») и ГОСТ Р 34.13-2015 / ГОСТ 34.13-2018 [2], а также в Рабочем предложении № 7801 *(RFC* 7801) Инженерного совета Интернета (*Internet Engineering Task Force*, *IETF*) [3] используются как для локального шифрования, так и для защиты каналов связи.

Современные информационные системы усложняются, в том числе, в сторону использования ресурсоэффективных платформ. Составляющие таких информационных систем представляют собой самостоятельные вычислительные узлы, выполняющие конкретные задачи. Такими узлами могут выступать: устройства сбора сведений (датчики температуры, влажности, электроэнергии, химические датчики, радиации и пр.), исполнимые устройства (устройства удалённого управления питанием, управления освещением, беспилотные летательные аппараты и пр.) и управляющие устройства (например терминалы управления оператора). В совокупности они образуют т. н. Интернет вещей (*Internet of Things*, далее – *IoT*) и т.н. промышленный Интернет вещей (*Industrial Internet of Things*, далее – *IIoT*). Они используются не только в системах «умный дом» в гражданском секторе, но и в автоматизированных системах управления технологическими процессами в промышленности, и в военном секторе в том числе.

Отличительной особенностью этих устройств является их ресурсоэффективность. Она фактически определяется выполнением задач устройством минимально необходимыми ресурсами. Она является фундаментальным требованием, потому что определяет два основных экономических па-

раметра: цену конечной продукции и её низкое энергопотребление. Это позволило построить *IoT*, число устройств которого в 2017 году превысило население планеты [4].

Требование ресурсоэффективности делает задачу реализации ЗП «Кузнечик» нетривиальной. В качестве примера макетной ресурсоэффективной платформы можно рассмотреть *Arduino UNO* на базе микроконтроллера *ATMega*328*P*. Её основные характеристики: 16 МГц частота, 32 КиБ микропрограмма и 2 КиБ оперативной памяти, разрядность 8 бит [5]. Согласно [1] минимальный массив данных ЗП «Кузнечик» составляет 928 байт. Это число не учитывает константы и операции в конечном поле $GF(2)/p(x)$. В результате само ЗП «Кузнечик» сложно размещать в такой платформе, не говоря уже о реализациях целевых алгоритмов, которые должно выполнять устройство.

Цель: исследовать возможность реализуемости российского алгоритма симметричного шифрования «Кузнечик» в ресурсоэффективных платформах и предложить такую реализацию.

Работа выполнена во исполнение:

- п.п. з) п. 23 и п.п. а) и б) п. 25 части IV Указа Президента России № 646 от 05.12.2016 [6] в направлении достижения стратегических целей обеспечения информационной безопасности России в области государственной и общественной безопасности, и в направлении достижения стратегических целей обеспечения информационной безопасности России в экономической сфере;
- п. 55, п. 56 и п.п. 12 и п.п. 13 п. 57 части IV Указа Президента России № 400 от 02.07.2021 [7] как часть государственной политики направленной на достижение цели обеспечения информационной безопасности России.

Откуда можно заключить, что массовое применение (не только в части официальных требований) ЗП «Кузнечик», взамен западных аналогов, в программных, аппаратных и программно-аппаратных средствах и системах, реализующих информационные технологии, является обязательным и приоритетным направлением развития отрасли информационных технологий в России.

Работа направлена на устранение следующих причин слабого использования ЗП «Кузнечик» среди разработчиков информационных систем в России:

1. Иррациональный необоснованный страх необходимости лицензирования деятельности при разработке, производстве, реализации и эксплуатации информационных систем на базе ЗП «Кузнечик», а так же иррациональный необоснованный страх необходимости сертификации разрабатываемого продукта.

2. Высокий порог вхождения при реализации ЗП «Кузнечик». Требуются специализированные узкие знания области дискретной математики, криптографии и промышленного программирования для разработки реализации ЗП «Кузнечик» промышленного уровня.

3. Необходимость проведения научных исследований с целью выведения потребительских характеристик (потребляемые ресурсы и производительность) реализаций на требуемый промышленный уровень, такой, что использование реализации в ресурсоэффективных платформах рыночно жизнеспособно.

4. Низкое число готовых реализаций промышленного уровня в репозиториях свободного программного обеспечения. В результате, или готовые разработки (т.н. «студенческие» разработки) не эффективны по производительности, или разработки промышленного уровня, но не являются свободными.

Предлагаемые решения и их испытания

В работе проведены исследования по оптимизации ЗП «Кузнечик». Разработаны модификации алгоритма [1] на основе предвычислений. **Математически доказаны** эквивалентность этих решений оригинальному алгоритму [1].

Существуют другие реализации ЗП «Кузнечик»: обычные [8], для *Python* [9], производительные для *OpenMP* [10], оптимизированные за счёт специальных инструкций процессоров [11–13]. Помимо этого ведутся исследования наращивания производительности за счёт специализированных инструкций процессора [14] или использования *NVIDIA GPU* [15]. Существуют отдельные исследования и разработки для аппаратных вычислителей [16, 17]. Указанные решения либо используют оригинальный алгоритм [1], либо высокопроизводительные платформы с его же реализацией, но не ресурсоэффективные.

В работе созданы промышленные реализации ЗП «Кузнечик» для *ATmega*328*P* и *AMD*64. Они размещены на Первой Российской Платформе для разработчиков *GitFlic* по адресу https://gitflic.ru/project/penta/isrdt-kuznechik/.

Основная кодовая база является свободным программным обеспечением, и останется свободным, предоставляемым по лицензии *Mozilla Public License Version* 2.0 с полным текстом которой можно ознакомиться по адресу https://mozilla.org/MPL/2.0/. Кодовая база системы тестирования является свободным программным обеспечением, предоставляемым по лицензии *The MIT License (MIT)* с полным текстом которой можно ознакомиться по адресу https://mit-license.org/. В отличие от *Global Public License, MPL* 2.0 не требу-

ет раскрытия исходных кодов всего проекта, а только лишь изменённой части предлагаемых решений «*MPL* заражает файл, а не проект». Поэтому их можно интегрировать как в свободные, так и в проприетарные проекты.

Реализации написаны на базе `stdint.h` на языке *C++* без использования специализированных библиотек и виртуализации классов. Поэтому могут свободно переноситься между различными компиляторами *C++*. Разработка произведена в операционной систем на базе ядра *Linux*. Они соответствуют основным требованиям к разработке промышленного программного продукта, а именно использования: системы контроля версий, системы модульного тестирования, анализа покрытия кода тестами, профилирования, документирования и распространения с помощью пакетных систем.

Испытания для *ATmega*328*P* (табл. 1) проводились на *Arduino UNO*.

Таблица 1

Результаты испытаний решения для *ATmega*328*P*

Ветка репозитория	Версия	Тест 512 байт, мксек	Объём микро-программы, байт	Объём ОЗУ, байт	Примечание
ATmega328P	0.0.0.0.0	1 284 004	4 350 (13%)	268 (13%)	Оригинальный
ATmega328P	0.0.3.0.0	63 742	10 714 (33%)	268 (13%)	Предвычисления

Испытания для *amd*64 (табл. 2) проведены на стенде с параметрами: процессор *Intel*(*R*) *Core*(*TM*) *i*7-7820*X CPU* @ 3.60*GHz*, 2 ядра; оперативная память 8 ГиБ; подкачка отключена; операционная система *Astra Linux Special Edition* 1.7.5.9; компилятор *gcc version* 8.3.0 (*AstraLinuxSE* 8.3.0-6).

Таблица 2

Результаты испытаний решения для *amd*64

Версия	Тест 1 КиБ, мксек	Тест 1 МиБ, мксек	Тест 100 МиБ, мксек	Объём констант, байт	Примечание
0.0.0.0.0	9 660	10 019 347	–	768	Оригинальный алгоритм
0.0.0.0.1	2 015	2 002 685	–		
0.0.0.0.2	1 857	1 904 539	–		
0.0.0.0.3	1 463	1 446 894	–		
1.0.3.0.0	164	174 723	17 420 806	4 352	Предвычисления
1.0.3.0.1	31	34 580	3 321 074		
1.0.3.0.2	34	33 659	3 336 607		
1.0.3.0.3	34	37 475	3 640 831		

Опираясь на табл. 1 и 2 можно оценить прирост производительности за счёт разработанных математических методов независимо от параметров испытательного стенда (замеры произведены в равных условиях):

- *ATmega*328*P:* версия 0.0.3.0.0: в **20,14 раз**.
- *AMD*64: версия 1.0.3.0.2: в **56,58 раз**.

Предложенные решения являются частью проекта «Инструменты исследований и разработок информационных систем» (*information systems research and development tools, isrdt*). Проект является полностью свободным и будет оставаться свободным. Согласно сорокалетним традициям поддержки свободного программного обеспечения приведём ссылки, по которым можно присоединиться к проекту и/или поддержать проект в любой форме.

Проект `isrdt-kuznechik`
https://gitflic.ru/project/penta/isrdt-kuznechik/

Telegram Подписаться на бесплатный телеграмм-канал (тут можно подписаться на частный телеграмм-канал с обсуждением)

Проект «Инструменты исследований и разработок информационных систем» (information systems research and development tools, isrdt) в части исследований и разработок реализаций ГОСТ Р 34.12-2015 / ГОСТ 34.12-2018, ГОСТ Р 34.13-2015 / ГОСТ 34.13-2018 isrdt-kuznechik выполнен при поддержке программы стратегического академического лидерства «Приоритет-2030» СибГУ им. М.Ф. Решетнёва, г. Красноярск.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ГОСТ 34.12–2018 Межгосударственный стандарт. Информационная технология. Криптографическая защита информации. Блочные шифры. Издание официальное. – М.: Стандартинформ, 2018. – 17 с.

2. ГОСТ 34.13–2018 Информационная технология. Криптографическая защита информации. Режимы работы блочных шифров. Издание официальное. – М.: Стандартинформ, 2018. – 28 с.
3. RFC 7801 GOST R 34.12-2015: Block Cipher "Kuznyechik" [Электронный ресурс]. – URL: https://datatracker.ietf.org/doc/html/rfc7801 (дата обращения: 24.02.2025).
4. Liam Tung IoT devices will outnumber the world's population this year for the first time [Электронный ресурс]. – URL: https://www.zdnet.com/article/iot-devices-will-outnumber-the-worlds-population-this-year-for-the-first-time/#ftag=RSSbaffb68 (дата обращения: 24.02.2025).
5. Arduino UNO R3. Product Reference Manual. SKU: A000066 [Электронный ресурс]. – URL: https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf (дата обращения: 24.02.2025).
6. Указа Президента России № 646 от 05.12.2016 «Об утверждении Доктрины информационной безопасности Российской Федерации» [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс».
7. Указ Президента России № 400 от 02.07.2021 «О стратегии национальной безопасности Российской Федерации» [Электронный ресурс]. Доступ из справ.-правовой системы «КонсультантПлюс».
8. Ковалев Роман Евгеньевич, Функции для шифрования/расшифрования данных в соответствии с ГОСТ Р 34.12-2015 (Кузнечик), реализованные на языке программирования Си [Электронный ресурс]. – URL: https://git.rekovalev.site/Crypto/C_Kuznechik_GOST_R_34.12-2015 (дата обращения: 24.02.2025).
9. Ilya Goldobin, kuznechik [Электронный ресурс]. – URL: https://github.com/ilyxenc/kuznechik (дата обращения: 24.02.2025).
10. Ajay Badrinath, Кузнечик шифр (GOST Kuznechik – 128) [Электронный ресурс]. – URL: https://github.com/AjayBadrinath/Kuznechik (дата обращения: 24.02.2025).
11. Семен Разенков, Шифрование на уровне протокола PDCP стандарта 3GPP [Электронный ресурс]. – URL: https://gitflic.ru/project/sirazenkov/3gpp-pdcp-crypto (дата обращения: 24.02.2025).
12. Semyon Dorokhin Grasshopper [Электронный ресурс]. – URL: https://github.com/svdprima/Grasshopper (дата обращения: 24.02.2025).
13. *Markku-Juhani O.* Saarinen kuznechik [Электронный ресурс]. – URL: https://github.com/mjosaarinen/kuznechik (дата обращения: 24.02.2025).
14. *Бородин М.А., Рыбкин А.С.* Высокоскоростные программные реализации блочного шифра «Кузнечик» // Проблемы информационной безопасности. Компьютерные системы. – 2014. – № 3. – С. 67-73.
15. *Удальцов В.А., Павлов В.Э.* Увеличение скорости работы алгоритма шифрования «Кузнечик» с использованием технологии CUDA // Теория. Практика. Инновации. – 2017. – № 4 (16). – С. 5-11.
16. *Калистру И.И., Бородин М.А., Рыбкин А.С., Гладько Р.А.* Способы реализации алгоритма «Кузнечик» на программируемых логических интегральных схемах // Радиопромышленность. – 2018. – Т. 28, № 3. – С. 64-70.
17. *Курганов Е.А.* О глубине аппаратной реализации блочного шифра кузнечик // Интеллектуальные системы. Теория и приложения. – 2016. – Т. 20, № 1. – С. 61-78.

УДК 004.738

С.Д. Ларин, А.Д. Минаев

Севастопольский государственный университет,
Россия, г. Севастополь,

МЕТОДЫ РЕКОНСТРУКЦИИ ТОПОЛОГИИ СЕТИ И МОДЕЛИРОВАНИЯ СЕТЕВЫХ СОБЫТИЙ ДЛЯ АНАЛИЗА КИБЕРИНЦИДЕНТОВ

В статье рассматриваются методы реконструкции топологии сети и моделирования сетевых событий, применяемые для анализа и расследования киберинцидентов. Целью работы является систематизация существующих подходов, выявление их преимуществ и недостатков, а также формирование рекомендаций по выбору оптимальных решений для построения достоверной картины инцидента в информационно-телекоммуникационных сетях. В рамках исследования был проведён комплексный анализ протоколов и признаков, позволяющих идентифицировать узлы, определить их роль в сети и восстановить логические связи между ними на основе PCAP-файлов. Особое внимание уделено анализу сетевого трафика на различных уровнях модели OSI – от канального до прикладного. Рассмотрены возможности таких инструментов, как Wireshark, Scapy, Pyshark и NetworkX, обеспечивающих автоматизацию анализа, построение графов взаимодействий и визуализацию сетевой структуры. В части моделирования сетевых событий описаны три ключевых подхода: сценарное, имитационное и основанное на генерации искусственного трафика. Представлены примеры использования таких платформ, как GNS3, EVE-NG, Mininet, а также инструментария Kali Linux, Metasploit и TRex. Показано, что интеграция методов реконструкции и моделирования позволяет формировать целостное представление об атаке, определить её хронологию, ключевые этапы и вовлечённые элементы инфраструктуры. Полученные результаты могут быть применены в системах мониторинга, киберполигонов и учебных стендов для подготовки специалистов в области информационной безопасности. Статья ориентирована на исследователей и специалистов, занимающихся анализом сетевых угроз и разработкой средств реагирования на киберинциденты, а также на студентов, изучающих сетевую безопасность.

Ключевые слова: *реконструкция топологии сети, киберинциденты, PCAP-анализ, моделирование сетевых событий, информационная безопасность, имитационное моделирование.*

The article discusses the methods of network topology reconstruction and modeling of network events used for the analysis and investigation of cyber incidents. The purpose of the work is to systematize existing approaches, identify their advantages and disadvantages, and make recommendations on choosing optimal solutions to build a reliable picture of an incident in information and telecommunications networks. As part of the

study, a comprehensive analysis of protocols and features was carried out to identify nodes, determine their role in the network, and restore logical connections between them based on PCAP files. Special attention is paid to the analysis of network traffic at various levels of the OSI model – from channel to application. The possibilities of tools such as Wireshark, Scapy, Pyshark, and NetworkX, which automate analysis, graph interactions, and visualize network structure, are considered. In terms of modeling network events, three key approaches are described: scenario-based, simulation-based, and artificial traffic generation. Examples of using platforms such as GNS3, EVE-NG, Mininet, as well as Kali Linux, Metasploit, and TRex tools are presented. It is shown that the integration of reconstruction and modeling methods makes it possible to form a holistic view of the attack, determine its chronology, key stages and the infrastructure elements involved. The obtained results can be applied in monitoring systems, cyber polygons and training stands for training specialists in the field of information security. The article is aimed at researchers and specialists involved in the analysis of network threats and the development of means of responding to cyber incidents, as well as students studying network security.

Keywords: *network topology reconstruction, cyber incidents, PCAP-analysis, network event modeling, information security, simulation modeling.*

Введение

В условиях стремительного развития информационных технологий и повсеместной цифровизации киберпреступления приобретают всё более сложный характер. Современные злоумышленники используют продвинутые методы сокрытия следов, а их атаки становятся более изощрёнными и труднодоступными для традиционных методов обнаружения. В связи с этим возрастает необходимость разработки эффективных инструментов анализа и расследования киберинцидентов. Одним из ключевых направлений в этой области является анализ сетевого трафика [1], который позволяет выявлять подозрительные активности, устанавливать последовательность событий и восстанавливать топологию сети для последующего выявления аномалий.

Однако традиционный анализ трафика зачастую ограничивается декодированием пакетов, выявлением аномалий и агрегацией логов. Эти решения фокусируются на отдельных аспектах анализа, оставляя за рамками своего функционала задачи реконструкции сетевой топологии, автоматической классификации устройств по их ролям (серверы, маршрутизаторы, клиенты) и интеграции с моделями симуляции событий. В результате исследователи сталкиваются с трудностями при построении целостной картины инцидента, особенно в условиях распределённых инфраструктур, где критически важны понимание связей между узлами и их функционального назначения.

Для успешного расследования киберинцидентов необходимо не только фиксировать активность на уровне отдельных пакетов, но и реконструировать динамику взаимодействия устройств. Это требует применения специализированных методов анализа, способных выделить характерные признаки сетевых элементов (IP/MAC-адреса, протоколы, TTL-значения) и восстано-

вить структуру сети. При этом интеграция таких методов с моделированием событий в контролируемых средах (киберполигонах) открывает новые возможности: симуляция атак (DDoS, MITM, сканирование портов) позволяет воспроизвести эволюцию инцидента, определить ключевые этапы атаки и протестировать меры защиты.

1. Методы реконструкции топологии сети

1.1. Протоколы и признаки для идентификации узлов и построения связей

Для восстановления топологии сети по данным, извлечённым из PCAP-файлов, ключевое значение имеет анализ протоколов на различных уровнях модели OSI. Однако не все уровни предоставляют релевантные данные: в практическом анализе сетевого трафика преимущественно используются канальный (L2), сетевой (L3), транспортный (L4) и прикладной (L7) уровни. Уровни 1 (физический), 5 (сеансовый) и 6 (представления) либо недоступны в PCAP-файлах, либо не несут значимой информации для восстановления топологии.

Каждый уровень даёт специфические признаки, которые могут быть использованы для идентификации узлов, определения их роли в сети и построения связей между ними. Ниже представлены основные протоколы, применяемые в рамках каждого уровня, и признаки, извлекаемые из них:

1. Канальный уровень (Data Link Layer) [2, 3]:

- ARP (Address Resolution Protocol) – используется для сопоставления IP-адресов и MAC-адресов в пределах локальной сети. Анализ ARP-запросов и ответов позволяет выявить активные хосты и построить таблицы соответствия IP и MAC. Это важно для определения физических устройств и их положения в сегментах локальных сетей. Частота ARP-запросов может указывать на роль узла (например, шлюз или маршрутизатор);
- LLDP (Link Layer Discovery Protocol) и CDP (Cisco Discovery Protocol**)** – протоколы обнаружения соседних устройств. Используются для определения физической связности сетевых устройств на уровне канала передачи. Благодаря данным протоколам происходит идентификация связей между маршрутизаторами, коммутаторами и конечными узлами.

2. Сетевой уровень (Network Layer) [2, 4]:

- IP (Internet Protocol) – основной источник информации об адресах источника и получателя трафика. Использование IP-адресов позволяет идентифицировать узлы и оценивать логические маршруты передачи данных, а также определять подсети и их границы. Значение TTL (Time-to-Live) – используется для оценки расстояния до хоста;
- ICMP (Internet Control Message Protocol) – диагностический протокол. ICMP-пакеты позволяют построить маршрутную карту сети, выявить активные хосты, определить недоступные узлы и проблемы маршру-

тизации. ICMP Echo Request / Reply – индикаторы активности хоста (ping). ICMP Destination Unreachable – свидетельство отсутствия маршрута. ICMP Time Exceeded – используется в traceroute для определения пути;

- OSPF (Open Shortest Path First) и BGP (Border Gateway Protocol) – протоколы маршрутизации. Анализ их пакетов позволяет выявить маршрутизаторы и логические связи между сегментами сети. OSPF Hello и LSA-сообщения – данные о соседях и структуре маршрутов. BGP UPDATE-сообщения – позволяют определить взаимодействие между автономными системами.

3. Транспортный уровень (Transport Layer) [4, 5]:

- TCP (Transmission Control Protocol) – установление соединений между клиентами и серверами. Анализ TCP-сессий позволяет определить направления и инициаторов взаимодействий. Множество входящих соединений – индикатор серверной роли узла. Частота инициирования TCP-сессий – указывает на клиентскую активность. Используемые порты (80, 443, 22 и др.) позволяют определить тип службы;
- UDP (User Datagram Protocol) – применяется в сервисах без установления соединения (DNS, SNMP, DHCP). Характеризуется высокой скоростью передачи и меньшей надёжностью. Частота и типы портов указывают на запущенные сервисы. Аномально высокая активность может свидетельствовать о сканировании или DDoS-атаках;
- SCTP (Stream Control Transmission Protocol) – реже используемый, но актуален для телекоммуникационных сетей (например, в сетях LTE и 5G).

4. Прикладной уровень (Application Layer) [2, 3, 6, 7]:

- DNS (Domain Name System) – обеспечивает разрешение имён в IP-адреса. Используется для классификации устройств;
- DHCP (Dynamic Host Configuration Protocol) – позволяет выявить роли устройств (DHCP-сервер, клиент), определить их MAC- и IP-адреса, а также длительность аренды;
- SNMP (Simple Network Management Protocol) – может использоваться для получения конфигурационной информации о сетевых устройствах;
- HTTP/HTTPS, FTP, SSH и др. – протоколы прикладного уровня предоставляют информацию о предоставляемых сервисах, что помогает классифицировать устройство (например, веб-сервер, файловый сервер и т.д.);
- NTP (Network Time Protocol) – протокол синхронизации времени. По нему можно определить серверы времени и связность между узлами по временным меткам;
- SMB (Server Message Block) – протокол обмена файлами, часто используется в Windows-сетях. Наличие трафика SMB указывает на файловые серверы.

1.2. Использование данных из PCAP-файлов

PCAP-файлы (Packet Capture) представляют собой стандартный формат хранения захваченного сетевого трафика, содержащий последовательность сетевых пакетов с их метаданными. Эти файлы широко применяются в задачах анализа, диагностики, мониторинга и расследования сетевых инцидентов. В контексте реконструкции топологии сети PCAP-файлы являются основным источником эмпирических данных, позволяющих восстановить структуру взаимодействий между узлами, выявить логические связи, а также определить роли устройств в информационно-телекоммуникационной системе.

Каждая запись в PCAP-файле содержит заголовок, фиксирующий временную метку захвата, размер пакета и служебную информацию, а также полезную нагрузку, включающую данные канального, сетевого, транспортного и прикладного уровней модели OSI. Таким образом, анализ PCAP-файлов позволяет получать данные сразу с нескольких уровней сетевого стека, что делает возможной многоуровневую реконструкцию топологии.

Файлы формата PCAP являются стандартом хранения сетевого трафика, захваченного в процессе мониторинга сети. Они представляют собой бинарные структуры, содержащие данные о сетевых пакетах с временными метками, а также информацией с различных уровней модели OSI. Использование PCAP-файлов в процессе реконструкции топологии сети позволяет анализировать как статическую структуру взаимодействий между узлами, так и их динамику во времени.

Как отмечается в работе В.И. Маркина, структура PCAP-файла включает в себя глобальный заголовок, содержащий параметры формата, и последовательность записей с заголовками отдельных пакетов и их содержимым. Каждый заголовок пакета содержит временные метки (в секундах и микросекундах), длину сохраняемого пакета и его фактический размер [8]. Такая структура позволяет выполнять детальный временной анализ сетевой активности.

Каждый пакет в PCAP сопровождается меткой времени, что делает возможным хронологический анализ сетевых событий. Это особенно важно при расследовании киберинцидентов, где важно установить точное время установления соединений, передачи данных или появления аномалий. Как показано в [8], временные характеристики используются при восстановлении TCP-сессий, а также при построении графов активности в распределённых системах мониторинга.

На основе анализа IP- и MAC-адресов, извлекаемых из заголовков Ethernet и IP-протокола, возможно определение уникальных узлов в сети. Для повышения точности сопоставления адресов часто используется информация из ARP-пакетов. Также возможно использование прикладных протоколов, таких как DHCP, для выявления ролей устройств (например, DHCP-серверов и клиентов) [9].

Протокол ICMP позволяет определить активность узлов при помощи сообщений Echo Request / Reply, а также выявить проблемы маршрутизации на основе сообщений Destination Unreachable и Time Exceeded [9].

TCP и UDP-сессии, идентифицируемые по комбинации IP-адресов и портов источника/назначения, дают основу для построения направленного графа взаимодействий между узлами. TCP-потоки позволяют чётко установить направление связи, в том числе определить инициатора соединения на основе анализа трёхстороннего рукопожатия SYN–SYN/ACK–ACK [8]. В свою очередь, анализ UDP-потоков может дать информацию о службах, использующих этот протокол, включая DNS, SNMP и другие [10].

Протоколы прикладного уровня (HTTP, DNS, SMB и др.) позволяют выявить функциональные роли устройств, например, веб-серверы, файловые хранилища, принтеры. В частности, анализ HTTP-заголовков и DNS-запросов позволяет дополнительно идентифицировать устройство по доменному имени или URL-адресу запроса [11]. DHCP-запросы дают информацию о сроках аренды IP-адресов, MAC-адресах клиентов и идентификаторах DHCP-серверов [9].

Несмотря на широкие возможности, анализ PCAP-файлов сопровождается рядом ограничений:

- данные могут быть неполными при захвате трафика только на отдельных узлах сети;
- NAT и VPN искажают реальную топологию и адресацию;
- использование шифрования (например, HTTPS) ограничивает анализ содержимого прикладного уровня.

1.3. Инструменты и библиотеки для анализа сетевого трафика

Современные задачи анализа сетевого трафика, включая реконструкцию топологии сети, требуют применения специализированных инструментов и библиотек. Их использование позволяет автоматизировать обработку PCAP-файлов, анализировать заголовки пакетов различных уровней модели OSI, выявлять закономерности в коммуникациях между узлами и выявлять аномалии, свидетельствующие о киберинцидентах.

Инструменты анализа сетевого трафика условно делятся на:

- прикладные программы с графическим интерфейсом (GUI);
- консольные утилиты;
- библиотеки для программирования на высокоуровневых языках (Python, Java, C и др.).

Wireshark – наиболее известный анализатор сетевого трафика с графическим интерфейсом. Он предоставляет удобные средства для визуального изучения пакетов, фильтрации по протоколам и IP-адресам, анализа потоков TCP/UDP, а также декодирования заголовков более 1000 протоколов.

Для автоматизации используется консольный аналог – TShark, который позволяет экспортировать данные в текстовый или XML-формат для последующей обработки. Как указано в [8], Wireshark остаётся ключевым инструментом при ручном исследовании содержимого PCAP-файлов.

Scapy – это гибкая и мощная библиотека на языке Python, предназначенная для создания, отправки, захвата и анализа сетевых пакетов. Она позволяет производить декомпозицию пакетов по уровням OSI, извлекать данные из заголовков, а также моделировать сетевой трафик.

Как отмечено в [11], Scapy используется при разработке собственных утилит анализа трафика и позволяет решать задачи по идентификации узлов, анализу направлений трафика и восстановлению связей между устройствами.

Pyshark – это высокоуровневая обёртка над TShark для Python. Она позволяет быстро и удобно парсить PCAP-файлы, получать доступ к полям различных протоколов и строить структурированные представления пакетов. В работе [11] указано, что библиотека применяется для извлечения ключевых признаков сетевых событий, включая анализ DNS, HTTP, DHCP и других протоколов прикладного уровня.

Для задач распределённого анализа сетевого трафика на платформе Java используются библиотеки jNetPcap и Hadoop-pcap-lib. Первая основана на обёртке над libpcap с использованием Java Native Interface (JNI), что обеспечивает доступ к широкому спектру протоколов. Вторая – ориентирована на работу в среде Hadoop и применяется при обработке больших объёмов PCAP-данных в рамках параллельных вычислений.

В работе [10] описаны особенности работы этих библиотек, включая организацию потоковой обработки, разбор заголовков пакетов, интеграцию с MapReduce и представление результатов в виде графов взаимодействий.

Для построения графа взаимодействий и реконструкции топологии сети по извлечённым данным из PCAP-файлов активно применяется библиотека NetworkX. Она позволяет описывать структуру сети в виде графа, рассчитывать степени узлов, центральность, компоненты связности и визуализировать связи между хостами. Как указано в [8], NetworkX может быть интегрирована с результатами анализа Scapy или Pyshark для автоматического построения модели сети.

2. Методы моделирования сетевых событий

2.1. Сценарное моделирование

Сценарное моделирование представляет собой целенаправленное воспроизведение конкретных атак, соответствующих типовым шаблонам злоумышленников. Оно осуществляется вручную или с использованием полу-

автоматических средств в контролируемой среде. Наиболее распространёнными являются атаки типа DDoS, ARP-spoofing, Man-in-the-Middle, SQL-инъекции и атаки на веб-приложения [12].

Для реализации сценариев применяются такие инструменты, как Kali Linux, включающая широкий набор утилит пентестинга; Metasploit Framework для генерации эксплойтов; LOIC (Low Orbit Ion Cannon) – для моделирования DDoS-атак; hping3 – для генерации кастомизированных TCP/UDP/ICMP пакетов.

Преимуществом данного подхода является высокая степень достоверности, возможность детализированного управления этапами атаки, а также адаптация под конкретные цели моделирования. Однако к его ограничениям относятся необходимость экспертной подготовки и ограниченная масштабируемость в больших средах.

2.2. Генерация искусственного трафика

Метод генерации искусственного трафика основан на создании заранее подготовленных PCAP-файлов либо на синтетическом формировании пакетов, имитирующих легитимные или вредоносные действия. Данный подход позволяет многократно воспроизводить однотипные сценарии и использовать их для нагрузочного тестирования или обучения систем обнаружения атак.

Инструменты, применяемые в этой области, включают [13]:

- Ostinato – генератор пакетов с графическим интерфейсом, поддерживающий множество протоколов;
- Nping – это мощный инструмент из состава пакета Nmap, предназначенный для генерации сетевых пакетов и измерения характеристик сети.
- TRex – это высокопроизводительный генератор сетевого трафика от Cisco, ориентированный на тестирование производительности сетевого оборудования.
- NetScanTools Packet Generator – Коммерческий инструмент для создания и отправки настраиваемых пакетов.

Генерация трафика может применяться как для моделирования «нормального» поведения пользователей, так и для воспроизведения атак с заранее известными характеристиками. Это упрощает тестирование реакций системы на повторяющиеся угрозы, однако ограничивает реалистичность поведения атакующего.

2.3. Имитационное моделирование

Имитационное моделирование основывается на создании виртуальных сетей с использованием эмуляторов и симуляторов сетевой инфраструктуры. Наиболее популярными платформами являются [14]:

- GNS3 (Graphical Network Simulator-3) – визуальное моделирование маршрутизаторов и коммутаторов на основе реальных образов;
- EVE-NG (Emulated Virtual Environment Next Generation) – среда построения масштабных виртуальных стендов;
- Mininet – симулятор сетей на основе OpenFlow и SDN, широко применяемый в научных исследованиях.

Данный подход позволяет гибко настраивать топологии, эмулировать работу маршрутизаторов, коммутаторов, серверов и клиентов, а также отслеживать поведение трафика между узлами. Имитационное моделирование особенно ценно для тестирования взаимодействия протоколов, анализа маршрутов, генерации трафика и его захвата в формате PCAP для последующего анализа. К преимуществам относится масштабируемость, гибкость настройки и возможность интеграции с системами мониторинга и анализа. Ограничениями являются требовательность к ресурсам и необходимость технической подготовки.

Заключение

В ходе проведённого исследования были рассмотрены современные методы реконструкции топологии сети и моделирования сетевых событий, направленные на повышение эффективности анализа и расследования киберинцидентов. Анализ существующих подходов позволил выявить их ключевые преимущества, ограничения и области применения. Особое внимание было уделено возможностям извлечения признаков сетевой структуры из PCAP-файлов, автоматизации анализа при помощи специализированных инструментов и библиотек, а также методам воспроизведения атак в контролируемых условиях с целью тестирования систем защиты.

Установлено, что интеграция методов реконструкции топологии с моделированием сетевых событий позволяет не только более точно воссоздавать хронологию инцидентов, но и формировать реалистичные сценарии для обучения и тестирования. Применение имитационного и сценарного моделирования в совокупности с анализом захваченного трафика обеспечивает комплексный подход к изучению сетевой активности, выявлению уязвимостей и выработке стратегий реагирования на инциденты.

Полученные результаты могут быть использованы при разработке программных средств для мониторинга и визуализации сетевого взаимодействия, создании киберполигонов, а также в учебных и исследовательских целях. В дальнейшем планируется расширить функциональность программных инструментов за счёт автоматической классификации устройств, оценки рисков и интеграции с системами предиктивного анализа угроз.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Качаева Г.И., Шихметова З.М.* Особенности анализа сетевого трафика при расследовании киберпреступлений // Неделя науки - 2023: Сборник материалов 44 итоговой научно-технической конференции преподавателей, сотрудников, аспирантов и студентов ДГТУ, Махачкала, 17–29 апреля 2023 года. – Махачкала: Дагестанский государственный технический университет, 2023. – С. 83-84. – EDN BRSQTA.
2. *Юхимук Роман Алексеевич, Веревкин Сергей Александрович*. Анализ протоколов сетевого взаимодействия для повышения надежности, быстродействия и безопасности сети организации // Известия ТулГУ. Технические науки. – 2023. – № 8.
3. *Олифер Виктор, Олифер Наталья.* Компьютерные сети. Принципы, технологии, протоколы: Юбилейное издание. – СПб.: Питер, 2020. – 1008 с.: ил. – (Серия «Учебник для вузов»). – ISBN 978-5-4461-1426-9.
4. *Багиян Н.В., Беляев В.А., Тудвасев Д.А.* Анализ современных сетевых протоколов: технологии и тенденции // Молодежь. Образование. Наука. – 2024. – № 1 (19). – С. 328-332. – EDN VRVBKA.
5. *Косьянова М.С., Грибовский А.А., Гаврин В.А., Баринов Д.М.* Устройство и перспективы применения протокола SCTP // Современные тенденции инженерного образования : Сборник материалов Научно-практической конференции, Санкт-Петербург, 22 апреля 2022 года. – СПб.: Федеральное государственное казенное военное образовательное учреждение высшего образования "ВОЕННАЯ АКАДЕМИЯ СВЯЗИ ИМЕНИ МАРШАЛА СОВЕТСКОГО СОЮЗА С.М. БУДЕННОГО" МИНИСТЕРСТВА ОБОРОНЫ РОССИЙСКОЙ ФЕДЕРАЦИИ, 2022. – С. 160-166. – EDN YPPEWZ.
6. *Юдина Н.Ю., Бунеев И.А., Чайка М.О.* Автоматизация выгрузки данных на FTP сервер с созданием резервных копий // Подготовка кадров в условиях перехода на инновационный путь развития лесного хозяйства: Научно-практическая конференция, Воронеж, 21–22 октября 2021 года. – Воронеж: Воронежский государственный лесотехнический университет им. Г.Ф. Морозова, 2021. – С. 279-283. – EDN IHORNH.
7. *Чан Т.З., Ха Т.К.* Защита NTP-серверов от DDOS-атак // Информационные технологии в науке, управлении, социальной сфере и медицине: Сборник научных трудов Международной научной конференции, Томск, 29 апреля – 02 2014 года / Национальный исследовательский Томский политехнический университет. Том Часть 1. – Томск: Национальный исследовательский Томский политехнический университет, 2014. – С. 204-205. – EDN TJJNSV.
8. *Маркин Ю.В.* Методы и средства углубленного анализа сетевого трафика: дис. … канд. техн. наук. – М.: ИСП РАН, 2017. – 80 с.
9. *Егоров Е.В., Овчаров В.А.* Технология пассивной идентификации вредоносного DOS-трафика в задачах расследования инцидентов информационной безопасности // Информационная безопасность. – 2022. – № 4. – С. 121-125.
10. *Василишин Н.С., Ушаков И.А., Котенко И.В.* Исследование алгоритмов анализа сетевого трафика с использованием технологий больших данных для обнаружения компьютерных атак // Аллея науки. – 2018. – № 6 (22). – С. 1-10.
11. *Нешумов Е.В.* Обнаружение кибератак с помощью Python // Международный научно-исследовательский конкурс. – 2023. – С. 285-288.

12. *Барыбина А.З.* Моделирование угроз информационной безопасности сценарным подходом // ЕГИ. – 2022. – № 4 (42). – URL: https://cyberleninka.ru/article/n/modelirovanie-ugroz-informatsionnoy-bezopasnosti-stsenarnym-podhodom.
13. WAN and LAN Network Traffic Generators [Электронный ресурс] // NetAdminTools. – URL: https://www.netadmintools.com/wan-and-lan-network-traffic-generators/ (дата обращения: 13.03.2025).
14. Top Most Popular Network Simulation Tools [Электронный ресурс] // PyNet Labs. – URL: https://www.pynetlabs.com/top-most-popular-network-simulation-tools/ (дата обращения: 13.03.2025).

УДК 004.056

Ш.Ж. Мусиралиева, М.А. Болатбек, К.Д. Байсылбаева

МУЛЬТИКЛАССОВЫЙ ПОДХОД К ВЫЯВЛЕНИЮ ИДЕОЛОГИЧЕСКОГО КИБЕРЭКСТРЕМИЗМА НА КАЗАХСКОМ ЯЗЫКЕ

Статья посвящена разработке подхода к автоматической мультиклассовой классификации идеологических направлений киберэкстремистского контента на казахском языке с использованием методов искусственного интеллекта. В целях решения поставленной задачи сформирован специализированный корпус, который состоит из текстов, извлеченных из социальных сетей размеченных по четырём классам: нейтральный текст, пропаганда, рекрутмент и радикализация. Был проведён сравнительный анализ эффективности различных алгоритмов классификации, включая классические методы машинного обучения (Logistic Regression, Naive Bayes, Random Forest, SVM), нейросетевые архитектуры (LSTM) и трансформерные модели (BERT). По результатам экспериментов лучшей моделью была признана BERT с архитектурой LSTM, достигшая наибольших показателей точности и полноты (Accuracy и F1-score = 0,944). Были обсуждены особенности и ограничения анализа казахского языка, связанные с его морфологической сложностью. Практическая значимость исследования заключается в возможности использования разработанной методики для автоматизированного мониторинга и выявления идеологически мотивированных угроз в сегменте интернета Казахстана.

Ключевые слова: *мультиклассовая классификация, киберэкстремизм, идеологические направления, искусственный интеллект, BERT, социальные сети, экстремистский контент.*

The article focuses on developing an approach for the automatic multiclass classification of ideological directions in cyber-extremist content in the Kazakh language using artificial intelligence methods. To achieve this objective, a specialized corpus was created, consisting of texts extracted from social networks and annotated into four categories: neutral text, propaganda, recruitment, and radicalization. A comparative analysis was conducted to evaluate the effectiveness of various classification algorithms, including traditional machine learning methods (Logistic Regression, Naïve Bayes, Random Forest, SVM), neural network architectures (LSTM), and transformer-based models (BERT). According to the experimental results, the BERT model with an LSTM architecture demonstrated the highest accuracy and F1-score (0.944), making it the most effective approach. The study also discusses the linguistic challenges and limitations associated with analyzing the Kazakh language due to its morphological complexity and provides recommendations for further model improvements. The practical significance of this research lies in its potential application for automated monitoring and the detection of ideologically motivated threats in Kazakhstan's internet segment.

Keywords: *multiclass classification, cyber-extremism, ideological directions, artificial intelligence, BERT, Kazakh language, social networks, extremist content*.

Введение

Современное цифровое пространство стало одной из ключевых площадок для распространения идеологически заряженного контента, включая экстремистские и радикальные высказывания. Интернет и социальные сети предоставляют широкие возможности для вербовки, пропаганды и координации действий различных экстремистских групп. В последние годы наблюдается рост киберэкстремизма, который охватывает различные идеологические направления – от политического и религиозного радикализма до этнонационалистической пропаганды [1, 2].

Несмотря на активные исследования в области выявления экстремистского контента, большинство разработок сосредоточено на анализе текстов на английском, русском и других широко распространенных языках. В то же время экстремистская риторика активно адаптируется под региональные особенности, включая использование казахского языка. Однако автоматизированные методы анализа текстов на казахском языке остаются слабо изученными, что создает серьезные вызовы в области мониторинга киберугроз.

Данная проблема приобретает особую значимость в связи с обеспечением национальной безопасности и межэтнического согласия. Необходимость эффективного выявления и классификации экстремистского контента на казахском языке обусловлена стремлением противодействовать радикализации общества, предотвращать возможные угрозы терроризма и экстремистской пропаганды, а также обеспечивать информационную безопасность.

В этой связи разработка и применение методов искусственного интеллекта для автоматической классификации идеологического направления киберэкстремистских текстов на казахском языке является актуальной задачей. Создание такой системы позволит оперативно анализировать и фильтровать подозрительный контент, а также предоставлять экспертам инструменты для раннего выявления потенциальных угроз.

Проблематика киберэкстремизма и его идеологического многообразия

Одной из ключевых сложностей в борьбе с киберэкстремизмом является неоднородность и разнообразие идеологических направлений, которые он охватывает. Современные экстремистские материалы могут быть обусловлены политическими, религиозными, националистическими и другими мотивами, что существенно затрудняет их автоматическое выявление и классификацию. Для эффективного мониторинга угроз и своевременного предотвращения радикализации недостаточно просто выявить наличие экстремистского содержания – нужно также точно определить его идеологическое направление [1, 2].

Большая часть существующих систем искусственного интеллекта и алгоритмов анализа текста направлены, главным образом, на бинарную классификацию (экстремизм или не экстремизм). Однако такие подходы недостаточны, поскольку не учитывают внутренние различия между идеологиями и не позволяют точно описать характер угрозы. Более того, исследования, посвящённые мультиклассовой классификации экстремистских текстов на казахском языке, на данный момент практически отсутствуют, что создаёт существенный научный пробел в данной области.

Таким образом, разработка методов искусственного интеллекта, позволяющих эффективно классифицировать экстремистский контент по идеологическим направлениям на казахском языке, является не только актуальной научной задачей, но и важным шагом для усиления информационной безопасности и оперативного мониторинга угроз в киберпространстве.

Цель и задачи исследования

Целью данного исследования является разработка методики и модели на основе искусственного интеллекта для автоматической мультиклассификации идеологических направлений киберэкстремизма на казахском языке. Для достижения поставленной цели были сформулированы следующие задачи:

1. Разработка модуля сбора данных.
2. Создание корпуса для мультиклассификации идеологического направления экстремистского содержания.
3. Создание модели мультиклассификации, определяющей идеологическую направленность экстремистского содержания в текстовых публикациях социальных сетей и мессенджеров (пропаганда деструктивных религиозных течений, радикализация и вовлечение в экстремистские и террористические организации).
4. Создание моделей мультиклассификации для определения идеологической направленности экстремистского содержания в аудио-и видеопубликах социальных сетей и мессенджеров.

Обзор литературы

Статья [3] посвящена разработке методов автоматического выявления сообщений с целью вербовки в социальных сетях, используемых экстремистскими и террористическими организациями. Исследование подчеркивает, что интернет играет значительную роль в процессе радикализации, однако существующие подходы преимущественно ориентированы на выявление экстремистского контента в целом, а не на выявление рекрутментных сообщений. В рамках предложенной методики авторы используют датасет, который был собран с экстремистских онлайн-ресурсов, который прошел ручную разметку на наличие и отсутствие признаков вербовки. В целях классификации были

применены алгоритмы машинного обучения, включая Naive Bayes, SVM и Logistic Regression, с признаковыми пространствами, основанными на Bag-of-Words и TF-IDF. Результаты экспериментов продемонстрировали, что предложенный метод позволяет достичь высокой точности (более 80%) в определении текстов класса рекрутмент. Анализ выявил характерные языковые конструкции, которые используются для привлечения новых сторонников, а также основные сложности при детекции скрытых призывов, завуалированных посредством иронии, сарказма и эвфемизмов.

Авторы делают вывод о необходимости дальнейшего усовершенствования моделей, считают применение более сложных нейросетевых архитектур, таких как трансформеры, а также расширение спектра анализируемых параметров, включая контекстные зависимости и сетевые связи авторов сообщений правильными. В перспективе исследование может быть дополнено многоязычным анализом и адаптацией предложенных методов для различных социокультурных сред [3].

Статья [4] рассматривает применение алгоритмов социальных сетей в целях противодействия распространению экстремистского контента. Авторы подчеркивают, что современные методы машинного обучения и алгоритмы анализа данных позволяют не только выявлять радикальный контент, но и ограничивать его распространение за счет автоматизированной фильтрации и модерации. Особое внимание уделяется тому, что такие алгоритмы могут не только выявлять потенциально опасные материалы, но и способствовать распространению позитивных ценностей.

Исследование [4] уделяет внимание на необходимости постоянного совершенствования алгоритмов с целью своевременного определения новых стратегий, применяемых экстремистами. В статье приводится анализ механизмов работы таких платформ, как Telegram и Facebook, рассматриваются принципы функционирования их систем модерации, а также рассматриваются преимущества и ограничения таких подходов. Кроме того, детально анализируются этические и правовые аспекты использования технологий искусственного интеллекта в контексте противодействия киберэкстремизму, включая вопросы защиты персональных данных и соблюдения прав пользователей.

Авторы приходят к выводу, что алгоритмы социальных сетей могут стать эффективным инструментом в борьбе с распространением экстремистского контента. Однако они подчеркивают необходимость учета правовых и этических норм при их разработке и внедрении. В перспективе дальнейшие исследования должны быть направлены на интеграцию многостороннего анализа (текст, изображение, видео), что позволит повысить эффективность обнаружения и блокировки радикального контента. Также авторы отмечают важность разработки гибких механизмов регулирования, способных адаптироваться к новым угрозам и изменяющимся стратегиям экстремистских группировок [4].

Статья [5] рассматривает разработку интеллектуального фреймворка, предназначенного для автоматизированного анализа экстремистского контента, распространяемого через социальные сети. В рамках исследования применялись методы веб-скрапинга, включая инструменты Selenium и Tweepy, что позволяет извлекать данные из Twitter и сохранить их в базе MongoDB. Тексты которые были собраны прошли процесс предобработки, включающий лемматизацию и токенизацию, а затем были аннотированы экспертами для обеспечения высокой точности разметки. Авторы провели сравнительный анализ нескольких алгоритмов машинного обучения, включая Random Forest и SVM, а также трансформерную модель BERT. По полученным результатам BERT продемонстрировал наивысшие показатели эффективности в классификации экстремистского контента, достигнув Precision = 0.93, Recall = 0.86, F1-score = 0.89 и Accuracy = 91%. Это подтверждает значительное превосходство трансформерных моделей над традиционными алгоритмами машинного обучения в задачах анализа текстов экстремистского характера.

Дополнительно авторы выполнили анализ тональности текстов, выявив ключевые лексические маркеры экстремистской риторики. В числе выявленных проблем исследования названы ограниченность исходного корпуса данных и дисбаланс классов, что потенциально влияет на устойчивость модели к различным видам экстремистского дискурса. В заключении подчеркивается значимость расширения набора данных, а также необходимость дальнейшей разработки методов анализа, включая поддержку различных языков и интеграцию дополнительных аналитических инструментов.

Статья [6] представляет собой обзор существующих подходов для автоматической классификации веб-ресурсов, содержащих экстремистский контент. Авторы описывают ключевые этапы процесса веб-классификации: предварительную обработку текстов, извлечение терминов (Bag-of-Words, N-граммы, стемминг и семантические методы) и отбор признаков (с использованием алгоритмов, таких как ACO, PSO и Harmony Search). Особое внимание уделено предложению авторов использовать алгоритм оптимизации «Krill Herd» для улучшения процесса отбора признаков.

В статье были рассмотрены три алгоритма классификации: наивный Байес, метод опорных векторов (SVM) и многослойный персептрон. По выводам авторов, предложенные улучшения позволяют значительно повысить точность и эффективность выявления экстремистского контента в интернете и подчёркивают необходимость дальнейших исследований, особенно применительно к арабскому языку и аналогичным задачам веб-классификации [6].

Статья [7] посвящена обнаружению и классификации экстремистских и агрессивных текстов в интернет-пространстве с применением моделей глубокого обучения. Авторы исследовали два датасета, содержащие сообщения с

платформ Stormfront и Twitter, где экстремистские тексты были выделены и размечены для дальнейшего анализа. На этапе предобработки текстов авторы удалили лишние символы, ссылки и специальные символы, после чего провели разбиение данных на две ключевые категории: «экстремизм» и «не экстремизм». В качестве моделей для классификации использовались BERT, ALBERT и ROBERTA, среди которых ROBERTA показала наивысшую эффективность, достигнув 87% точности, полноты и F1-score. Данные результаты подтверждают, что трансформерные модели, обученные на больших объемах данных, способны более точно выявлять экстремистский контент по сравнению с традиционными методами обработки текстов.

В заключении авторы подчеркивают перспективность использования ROBERTA в задачах анализа экстремистского контента, также отмечают, что качество разметки данных остается важным моментом, который влияет на точность модели. Кроме того, особое внимание уделяется этическим аспектам применения подобных технологий, включая защиту данных и корректность определения экстремистского контента. Исследователи рекомендуют дальнейшую доработку методов аннотации данных и адаптацию подходов для более сложных языковых контекстов [7].

Статья [8] посвящена разработке автоматизированных методов выявления радикального и экстремистского контента в интернете с использованием обработки естественного языка (NLP). Авторы подчеркивают актуальность проблемы, обусловленную активным использованием цифровых платформ экстремистскими группировками для пропаганды и вербовки. В исследовании дан обзор различных методов классификации экстремистского контента, включая классические алгоритмы машинного обучения (наивный Байес, логистическая регрессия, метод опорных векторов). Рассматриваются анализ тональности, аффективный анализ и стилистический анализ текста, которые позволяют более точно выявлять намерения авторов сообщений. При проведении экспериментов авторы сформировали собственный датасет на казахском языке, собранный из социальной сети «ВКонтакте». Размеченные данные были распределены по двум классам: «экстремистский» и «не экстремистский». В ходе тестирования наилучшие результаты продемонстрировал алгоритм Gradient Boosting, использующий векторные представления Word2Vec. Данная модель достигла 89% точности и 86% F1-score, что подтверждает её эффективность в задачах автоматической классификации экстремистских сообщений.

Авторы указывают на необходимость расширения корпуса данных и повышения качества аннотации, поскольку размер и репрезентативность датасета напрямую влияют на точность работы алгоритмов. Также подчеркивается важность включения дополнительных признаков, таких как учет контекста, сетевых взаимодействий и семантических особенностей языка, что позволит улучшить эффективность классификации экстремистских текстов в будущем [8].

Статья [9] рассматривает разработку динамического метода для автоматического выявления экстремистских и преступных публикаций в социальных сетях. Авторы предлагают модель полуконтролируемого машинного обучения, способную эффективно идентифицировать вредоносный контент даже при ограниченном количестве размеченных данных. Основу предложенного подхода составляет графовая оптимизация, включающая метод максимального потока для выбора наиболее информативных сообщений, требующих аннотации.

Предложенная модель функционирует в многовидовом режиме, анализируя как текстовую составляющую сообщений, так и метаинформацию, включая URL-адреса. Для оценки эффективности подхода были использованы три независимых датасета: два с сообщениями из Twitter, связанными с деятельностью ИГИЛ, и один с форума Ansar Aljihad Network, содержащий данные, релевантные анализу онлайн-рекрутмента.

Результаты экспериментов показали, что предложенный метод достигает точности около 80%, при этом значительно сокращает объем необходимой ручной разметки. Достигается это за счет выбора наиболее значимых и характерных текстов, содержащих чувствительные и эмоционально окрашенные выражения, а также характерные URL-ссылки, которые часто используются в экстремистском контенте.

Авторы показывают преимущества графового подхода, позволяющего эффективно отбирать данные для аннотации, что делает процесс обучения модели более экономичным и быстрым. Исследование подтверждает, что предложенная методика может быть адаптирована к различным видам контента, обеспечивая стабильные результаты на различных наборах данных.

Методология исследования

Особенности анализа текстов на казахском языке

Казахский язык обладает сложной морфологией, что влияет на применение алгоритмов обработки естественного языка (NLP). Будучи морфологически богатым языком, он характеризуется множеством аффиксов, изменяющих значение слов, что усложняет их анализ и классификацию. Одно слово может нести несколько грамматических значений, создавая разнообразие словоформ и усложняя задачу определения семантики текста.

Из-за морфологических особенностей казахского языка традиционные методы обработки текста, такие как мешок слов (Bag-of-Words) и TF-IDF, сталкиваются с проблемой высокой размерности признакового пространства и разреженности данных. Это приводит к потере связности текста и усложняет задачу обучения моделей.

Одним из решений является использование стемминга и лемматизации, которые позволяют привести слова к базовой форме, тем самым снижая разреженность данных и улучшая качество классификации. В частности, методы морфологического анализа могут повысить точность представления текстов для дальнейшей обработки.

Ключевой проблемой в обработке казахского текста является ограниченность открытых корпусов и отсутствие развитых NLP-инструментов, по сравнению с английским или русским языками. Поэтому улучшение предобученных языковых моделей становится важной задачей.

Таким образом, для автоматизированной мультиклассификации текста идеологического направления на казахском языке необходимо учитывать морфологические особенности и применять современные подходы на основе глубокого обучения. Использование трансформеров, таких как BERT, в сочетании с методами морфологического анализа, является оптимальной стратегией для повышения точности анализа экстремистских текстов и эффективного решения задачи мониторинга контента.

Разработка модели мультиклассификации идеологических направлений

В целях решения поставленной задачи была выбрана мультиклассовая классификация текста, которая позволит выявлять экстремистский контент, а также различать его идеологическую принадлежность. Выделены четыре основных идеологических класса:

- Нейтральный текст.
- Пропаганда.
- Рекрутмент.
- Радикализация.

На вход модели подается текст на казахском языке, а на выходе модель определяет, к какому именно из указанных идеологических классов относится данный текст. Данная мультиклассовая классификация необходима, так как имеет практическую значимость определения типа угрозы для более точного реагирования и принятия решений со стороны мониторинговых служб.

В ходе работы разработан модуль сбора данных. Модуль сбора данных разработан с использованием технологий API для поиска данных в социальных сетях Телеграм, ВКонтакте, Твиттер, Youtube. В выбранных социальных сетях проанализировано более 400 групп с признаками деструктивных убеждений. В модуле сбора данных реализован функционал пополнения базы данных в соответствии со списком ключевых слов и на основе выбранного временного интервала, запроса идентификатора группы.

Создан корпус мультиклассификации экстремистского содержания и идеологической направленности. В целях определения идеологической экстремистской направленности, собирающей текстовые данные из

социальных сетей и новостных сайтов, были определены правила и категории экстремистских и нейтральных текстов. Составлены списки ключевых слов для каждого класса, правила, списки групп для сбора данных. Корпус был отсортирован вручную по правилам.

В результате аннотации классам были присвоены следующие обозначения: Propaganda (0), Radicalization (1), Recruitment (2), Neutral (3). Собранный корпус был разделен на наборы для обучения и тестирования в соотношении 80% и 20% с целью применения методов машинного обучения. В каждом классе собрано более 2000 текстов. К данным корпуса применялись такие алгоритмы препроцессинга, как токенизация, очистка от пунктуации, очистка от наиболее распространенных слов, очистка от стоп-слов. Составлена статистика и визуализация корпуса. Использовались алгоритмы и модели Word2vec, bag of words и n-gram, и корпус был подготовлен к машинному обучению.

Создана модель мультиклассификации идеологического направления экстремистского содержания в текстовых публикациях социальных сетей и мессенджеров (пропаганда деструктивных религиозных течений, радикализация и вовлечение в экстремистские и террористические организации). К данным корпуса применялись алгоритмы word2vec, tf-idf. Поскольку тексты в веб-ресурсах в основном находятся в неструктурированном состоянии и заполняются различными пользователями, существует множество орфографических ошибок. Поэтому в первую очередь был предложен метод на основе Spell Checker для исправления орфографических ошибок в казахском языке.

Упомянутый метод является очень полезной функцией любой поисковой системы. Самое простое решение – отсортировать все позиции словаря по мере увеличения редакционного расстояния и отображать только первые несколько позиций. В качестве редакционного расстояния можно взять расстояние Левенштейна.

Расстояние Левенштейна-показывает минимальное количество попыток ввода/удаления/изменения символов, необходимых для преобразования исходной строки в целевую строку. Для определения идеологического направления в тексте был проведен сравнительный анализ классических машинных алгоритмов, таких как Logistic Regression, KNN, SVM, Naive Bayes, Decision Tree, Random Forest, Gradient Boosting. На базе Spellcheker+Stemming+TF-idf+LSTM+BERT была создана новая модель.

Настройка гиперпараметров моделей

Для предобученной модели BERT (Bert-base-multilingual-uncased) была выполнена тонкая настройка (fine-tuning) с фиксированными гиперпараметрами:

вход = 128 слов или токенов,

линейная классификация = 4,

bert_model_name = 'Bert-base-multilingual-uncased',
num_classes = 4,
max_length = 128,
batch_size = 64,
num_epochs = 10,
learning_rate = 2e-5.

Кроме того, для решения задачи мультиклассификации идеологических текстов были объединены 2 алгоритма глубокого обучения (BERT и LSTM, Bert+linear). Модель принимает последовательность текста в качестве входных данных вместе с соответствующими длинами каждой последовательности для LSTM. Он встраивает текст (вложенный размер = 20), обрабатывает его через слой LSTM (размер = 64), передает последнее скрытое состояние через полностью добавленные слои с активациями ReLU и, наконец, использует сигмовидную активацию для получения единственного выходного значения.

Гиперпараметры объединенной модели:
вход = 128 слов или токенов,
BERT = 768,
LSTM = 256,
DropOut = 0.2,
linear Classification = 4,
bert_model_name = 'Bert-base-multilingual-uncased',
num_classes = 4,
max_length = 128,
batch_size = 64,
num_epochs = 20,
learning_rate = 2E-5.

В ходе экспериментов проводилась настройка глубины архитектуры, а также подбирались параметры DropOut и размерность скрытых слоев для достижения наилучшего баланса между точностью и обобщающей способностью моделей.

Эксперименты и результаты

Сравнительный анализ алгоритмов машинного обучения

Анализ показал, что Logistic Regression с CountVectorizer достигла наилучшей точности (accuracy = 0.9357) среди классических моделей машинного обучения. Random Forest с TF-IDF также продемонстрировал высокую производительность (accuracy = 0.9350), что указывает на эффективность ансамблевых методов для данной задачи. Однако метод SVC (Support Vector Classifier), основанный на TF-IDF, показал несколько более низкую точность, что может быть связано с особенностями структуры данных.

Алгоритм Naive Bayes, использующий Count Vectorizer, также показал высокий результат (accuracy = 0.9259), что подтверждает его применимость для текстовой классификации. Тем не менее, модели K-Nearest Neighbors (KNN) продемонстрировали более низкие метрики (accuracy = 0.7659), что свидетельствует о сложности данного метода для высокоразмерных данных.

Глубокие модели: BERT + LSTM

Глубокие архитектуры на основе BERT и LSTM обеспечили наилучшие результаты по всем метрикам. В частности, объединенная модель BERT + LSTM с исправлением орфографии и стеммингом достигла общей точности 94,4%. При этом наилучшая производительность была зафиксирована для класса "нейтральный", что указывает на высокую надежность модели в идентификации данного класса.

В свою очередь, выявление "радикализации" показало несколько более низкие результаты (F1-score = 0.9258), что может быть связано с особенностями языка и сложностью обнаружения завуалированных радикальных высказываний:

- Классические методы машинного обучения (Logistic Regression, Random Forest) показали высокую точность, однако трансформерные модели обеспечили более надежные результаты за счет учета контекста.
- Комбинированная архитектура BERT + LSTM продемонстрировала наилучшие результаты, подтверждая эффективность трансформеров для обработки казахского текста.

Таким образом, результаты исследования подтверждают, что современные глубокие модели, включая BERT и LSTM, являются наиболее перспективными для решения задач мультиклассовой классификации идеологического контента.

Значение исследования для реальной практики

Разработанные модели имеют значительное практическое применение, поскольку они могут использоваться для мониторинга социальных сетей и других онлайн-платформ с целью выявления экстремистского контента на казахском языке. Внедрение этих моделей позволит автоматизировать процесс обнаружения потенциальных угроз, тем самым помогает облегчить работу специалистов, которые занимаются анализом информации и обеспечением кибербезопасности.

Мультиклассовый подход к классификации текста позволяет более точно определить идеологическую направленность экстремистских материалов, что способствует более эффективному противодействию различным видам угроз. Данная система способна выявлять признаки религиозного экстремизма, предотвращать распространение радикальных идей и распознавать попытки вербовки в экстремистские организации.

Заключение

В статье представлена методика автоматической мультиклассовой классификации идеологических направлений экстремистского контента на казахском языке с использованием современных методов искусственного интеллекта. В ходе исследования был подготовлен и экспертно размечен специализированный датасет, включающий тексты из социальных сетей и онлайн-форумов по категориям: нейтральный контент, пропаганда, рекрутмент и радикализация. Проведённый сравнительный анализ алгоритмов показал, что трансформерные модели, особенно BERT с архитектурой LSTM, демонстрируют наилучшие результаты, превосходя классические методы машинного обучения и простые нейронные сети благодаря способности учитывать сложные контекстные и морфологические особенности казахского языка.

Полученные результаты подтвердили, что предложенные подходы могут эффективно использоваться для автоматизированного выявления и мониторинга экстремистских материалов, тем самым содействуя усилению национальной и информационной безопасности Казахстана. В то же время в ходе работы были выявлены некоторые ограничения, связанные с размером и дисбалансом данных, а также спецификой языковой обработки агглютинативного казахского языка.

Перспективы дальнейших исследований включают несколько направлений. Первое из них расширение объёма корпуса и количества классов. Далее улучшение интерпретируемости моделей, а также совершенствование методов предварительной обработки и балансировки данных с целью дальнейшего повышения качества и надежности классификации экстремистского контента. Создание модели мультиклассификации социальных сетей и мессенджеров, определяющей экстремистское (религиозный экстремизм, политический экстремизм, ксенофобия) содержание в текстовых, аудио, видео публикациях является еще одним направлением. В дополнение планируется создание гибридной модели выявления наиболее активных киберпропаганд в социальных сетях и мессенджерах, разработка модели, алгоритмов и методов выявления сообществ в социальных сетях на основе заданного набора параметров. В рамках исследования также рассматривается создание чат-бота с диалогом на казахском языке для консультирования по вопросам религии и разработка программного обеспечения, реализующего разработанные методы и модели.

Данное исследование было профинансировано Министерством науки и высшего образования Республики Казахстан в рамках гранта AP19676342 “ Мультиклассификация идеологических направлений киберэкстремизма на казахском языке методами искусственного интеллекта”.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Daniel Ball, Reza Montasari.* The Evolution of Terrorism in Digital Era: Cyberterrorism, Social Media, and Modern Extremism.
2. *Gregory Shuck.* Online Jihadism: Propaganda, Recruitment And Homegrown Radicalization.
3. *Jacob R Scanlon and Matthew S Gerber.* Automatic detection of cyber-recruitment by violent extremists.
4. *Khalaf Tahat, Mohammed Habes, Ahmed Mansoori, Noura Naqbi, Najia Al Ketbi, Ihsan Maysari, Dina Tahat, Abdulaziz Altawil.* Social media algorithms in countering cyber extremism: A systematic review.
5. *Ala Berzinji, Iraq Rawaz A, Alrahman F Muhammed, Danial Abdulkareem Muhammed*. Development of an Intelligence Gathering Framework for AnalysingCyber Extremism on Social Media Networks.
6. *Hassan Awad Hassan Al- Sukhni, Madihah Mohd Saudi, Azuan Ahmad.* A Review of Web Classifier Approach with Possible Research Direction to Detect Cyber Extremists.
7. *Sudhanshu Patel, K. Raja, J. Shiny Duela, Thomas M Chen, Mithileysh Sathiyanarayanan*. Identifying Cyber Extremism Sentiments using ROBERTA.
8. *Shynar Mussiraliyeva, Milana Bolatbek, Batyrkhan, Zhanar Medetbek, Gulshat Baispay, Ruslan Ospanov*. On Detecting Online Radicalization and Extremism Using Natural Language Processing.
9. *Sreyasee Das Bhattacharjee, Bala Venkatram Balantrapu, William Tolone, Ashit Talukder*. Identifying Extremism in Social Media with Multi-view Context-Aware Subset Optimization..

УДК 004.657

О.Ю. Сабинин

Санкт Петербургский политехнический университет Петра Великого,
Россия, г. Санкт-Петербург

ОРГАНИЗАЦИЯ УСКОРЕННОГО СТАТИСТИЧЕСКОГО МОДЕЛИРОВАНИЯ СИСТЕМ ЗАЩИТЫ ИНФОРМАЦИИ

В работе рассматриваются процедуры организации статистического моделирования систем защиты информации, позволяющие существенно сократить затраты времени на оценку показателей эффективности систем защиты с заданной точностью. Предлагается адаптивная процедура определения границ слоев и процедура оптимального распределения выборки по слоям при одновременной оценке нескольких показателей системы защиты информации. Предлагаемые подходы к организации статистического моделирования сложных систем защиты информации позволяют в несколько раз сократить затраты машинного времени на проведение исследований.

Ключевые слова: *статистическое моделирование, процедура организации моделирования, оценки вероятностных характеристик.*

The paper considers the procedures for organizing statistical modeling of information security systems, which can significantly reduce the time spent on evaluating the effectiveness of security systems with a given accuracy. An adaptive procedure for determining the boundaries of layers and a procedure for optimal distribution of the sample by layers while simultaneously evaluating several indicators of the information security system are proposed. The proposed approaches to the organization of statistical modeling of complex information security systems make it possible to reduce the cost of machine time for conducting research several times.

Keywords: *statistical modeling, planning procedure, estimates of probabilistic characteristics.*

Введение

К настоящему времени разработано и апробировано на практике достаточно много методов моделирования систем защиты информации, позволяющих эффективно решать задачи анализа и синтеза систем защиты самой различной природы и архитектуры, но поскольку на процессы защиты информации решающее влияние оказывает большое число случайных факторов, основным методом исследования может служить метод статистического моделирования [1–3].

Положительные особенности этого метода – простота реализации и возможность использования для анализа систем практически любой сложности.

Однако метод статистического моделирования имеет принципиальный недостаток – низкую скорость сходимости. Поэтому для достижения требуемой точности результата может потребоваться проведение большого числа испытаний, что, как правило, приводит в силу сложности математического описания моделей к большим затратам машинного времени.

Постановка задачи

Целью статистического моделирования системы защиты информации является оценка эффективности защиты информации в системе.

В качестве показателей эффективности защиты могут использоваться различные показатели, например, вероятность своевременного обнаружения вредоносной программы и своевременного пресечения ее распространения на этапе инфицирования, среднее время между инцидентами возникновения угроз заражения вирусом и др. [1].

Несмотря на разнообразие показателей качества, используемых при оценке эффективности защиты информации, большинство показателей представляют собой усредненное значение какой-либо характеристики и задача оценки показателей качества, характеризующих эффективность системы защиты информации, эквивалентна вычислению совокупности многомерных интегралов [4]:

$$J^{(i)} = \int_{\Omega} \Phi_i(y) f(y) dy \quad (i = \overline{1,R}), \tag{1}$$

где $J^{(1)}, J^{(2)}, \ldots, J^{(R)}$ – оцениваемые показатели качества системы защиты информации; R – количество оцениваемых при статистическом моделировании характеристик,

$\Phi_i(Y)$ – оператор модели исследуемого процесса по выходной координате Z_i; f(y) – функция плотности вероятностей вектора случайных величин Y; Ω – пространство возможных значений вектора Y.

Статистическое моделирование по традиционной схеме основывается на использовании процедуры простой случайной выборки, при которой случайные величины модели Y реализуются в соответствии с функцией плотности вероятностей f(y), а оценки вероятностных характеристик $J^{(i)}$ производятся по соотношению:

$$\overline{J^{(i)}} = \frac{1}{n}\sum_{k=1}^{n} Z_k = \frac{1}{n}\sum_{k=1}^{n} \Phi_k \ (i=1,R),$$

где n – объем выборки.

При таком подходе к организации статистического моделирования для получения требуемой точности оценки может потребоваться очень много времени так, как увеличение точности оценки в s раз требует увеличения количества воспроизведений исследуемого процесса в s^2 раз.

Процедура организации выборки

Значительного уменьшения дисперсии оценки, а, следовательно, и сокращения затрат машинного времени по сравнению с простой случайной выборки можно добиться, используя специальные методы организации и планирования статистического моделирования, в основе которых лежит идея изменения закона распределения случайных величин таким образом, чтобы сделать результаты моделирования более однородными, не изменяя при этом математическое ожидание оцениваемых величин [5].

Рассмотрим организацию ускоренного статистического моделирования при помощи разбиения пространства возможных значений случайных величин модели Ω на L подмножеств (слоев) $\Omega_1, \dots . \Omega_L$ и проведения в каждом слое простой случайной выборки объема n_h [6].

В качестве оценки вероятностной характеристики в этом случае используется величина:

$$\bar{J}_p = \sum_{h=1}^{L} \omega_h \, \bar{J}_h,$$

где ω_h – вероятность попадания вектора Y в слой Ω_h; $\bar{J}_h$– оценка математического ожидания величины Z, характеризующей исход исследуемого процесса при выборе Y из слоя Ω_h

При соответствующем выборе $n_1, \dots . n_L$ дисперсия оценки может быть сделана меньшей, чем при простой случайной выборке при том же числе испытаний.

Процедура распределения выборки по слоям

При организации статистического моделирования системы при помощи разбиения выборки на слои следует учитывать, что [4]:

1) время воспроизведения исследуемого процесса зависит от того, какому слою принадлежат реализованные значения случайных величин,
2) требуемые точности (дисперсии) оценок различных оцениваемых характеристик могут различаться.

В таких условиях планирование выборки при оценке нескольких вероятностных характеристик целесообразно производить исходя из условия минимизации функции

$$T = \sum_{h=1}^{L} T_h n_h \qquad (2)$$

при ограничениях

$$D^{(i)} \le D^{(i)}_{з\,ад} \qquad (i=\overline{1,R}), \qquad (3)$$

где T – время, необходимое на получение оценок R характеристик с заданными дисперсиями $D^{(i)}_{з\,ад}$ $(i=\overline{1,R})$; T_h – время, необходимое на воспроизведение исследуемого процесса при розыгрыше вектора случайных величин в слое Ω_h; n_h – объем выборки в слое Ω_h; $D^{(i)}$ – дисперсия оценки характеристики $J^{(i)}$.

Наибольшие трудности при планировании статистического моделирования с одновременной оценкой нескольких вероятностных характеристик вызывает решение задачи распределения выборки по слоям.

Если выразить критерий оптимальности (2) и ограничения на дисперсии оценок (3) как функции аргумента

$$x_h = 1/n_h \qquad (h=\overline{1,L}), \qquad (4)$$

то задача оптимального распределения выборки по L слоям может быть представлена как задача определения значений x_h, обеспечивающих минимум функции

$$T = \sum_{h=1}^{L} T_h / x_h \qquad (5)$$

при условии

$$\sum_{h=1}^{L} a_{ih} x_h \le D^{(i)}_{з\,ад} \qquad (i=\overline{1,R}), \qquad (6)$$

где $a_{ih} = \omega_h^2 D^{(i)}_h$ $(h=\overline{1,L})$; $T_h \ge 0$; $D^{(i)}_{з\,ад} > 0$; $D^{(i)}_h \ge 0$; T_h – время, необходимое на воспроизведение исследуемого процесса при розыгрыше вектора случайных величин в слое Ω_h; $D^{(i)}_{з\,ад}$ – заданная дисперсия оценки i-й вероятностной характеристики; $D^{(i)}_h$ – дисперсия выходной координаты системы Z_i при выборе Y из слоя Ω_h; ω_h – вес слоя Ω_h.

При такой формулировке оптимальное распределение выборки по слоям представляет собой задачу нелинейного программирования, поскольку соотношение (5) определяет поверхность в L-мерном пространстве, а условие (6) при знаках равенства – плоскости размерности (L - 1).

Решением задачи являются L координат точки соприкосновения функции (5) с многогранником допустимых решений, определяемым условиями (6).

Для определения оптимальной точки может быть использована следующая процедура.

1. Находим координаты точки соприкосновения i-ой гиперплоскости с целевой поверхностью по формуле

$$x_{ih} = \frac{D_{\text{з ад}}^{(i)} \sqrt{T_h a_{ih}}}{a_{ih} \sum_{h=1}^{L} \sqrt{T_h a_{ih}}} \qquad (h = \overline{1,L};\quad i = \overline{1,R}). \tag{7}$$

Найденное таким образом решение $x_i = (x_{i1}, x_{i2}, \ldots, x_{iL})$ обеспечивает достижение заданной дисперсии $D_{\text{з ад}}^{(i)}$ за минимальное время T и позволяет на основе соотношения (4) определить оптимальное распределение выборки по слоям для i-й характеристики.

2. Выбираем некоторое частное значение i^* из множества {1,2,...,R} и проверяем, попадает ли точка $x_{i^*} = (x_{i_1^*}, x_{i_2^*}, \ldots, x_{i_L^*})$ в область допустимых решений, для этого должны выполняться неравенства

$$\sum_{h=1}^{L} a_{i_h} x_{i_h^*} \le D_{\text{з ад}}^{(i)} \qquad (i = \overline{1,R}). \tag{8}$$

Если все неравенства выполняются, то полученное решение является оптимальным.

3. Если условие (8) не выполняется для некоторых i, то следует выбрать новое значение i^* из множества тех i, для которых неравенство (8) оказалось несправедливым. После этого проверяем справедливость неравенства для нового значения i^*.

Таким образом, всегда можно найти оптимальное решение, если функция (5) касается только одной из плоскостей.

В случае, когда целевая функция соприкасается с пересечением k плоскостей, можно показать, что оптимальные значения x_h (h = 1,L) могут быть получены из уравнений

$$x_h' = \frac{\sqrt{T_h}\,(D_{\text{з ад}}^{(i_1)} - \lambda_1 D_{\text{з ад}}^{(i_2)} - \ldots - \lambda_{k-1} D_{\text{з ад}}^{(i_k)})}{\sqrt{(a_{i_1 h} - \lambda_1 a_{i_2 h} - \ldots - \lambda_{k-1} a_{i_k h})} \sum_{h=1}^{L} \sqrt{T_h (a_{i_1 h} - \ldots - \lambda_{k-1} a_{i_k h})}}.$$

Здесь $\lambda_1, \ldots, \lambda_{k-1}$ – параметры, удовлетворяющие системе уравнений:

$$\sum_{h=1}^{L} \frac{\sqrt{T_k} \sum_{j=1}^{k} \left[-\lambda_{j-1} (D_{зад}^{(i_j)} A_{i_d h} - a_{i_j h} b_{i_d}) \right]}{\sqrt{-\sum_{j=1}^{k} \lambda_{j-1} a_{i_j h}}} = 0, \qquad (11)$$

где

$$A_{i_d h} = a_{i_d h} - a_{i_{d+1} h} \qquad (d = \overline{1,(k-1)}; \qquad h = \overline{1,L});$$

$$b_{i_d} = D_{зад}^{(i_d)} - D_{зад}^{(i_{d+1})} \qquad (d = \overline{1,(k-1)});$$

$$\lambda_0 = -1.$$

Адаптивный алгоритм определения границ слоев

Наиболее сложной задачей, возникающей при планировании выборки на основе разбиения ее между слоями, является определение границ слоев.

Одним из возможных подходов к определению границ слоев при статистическом моделировании систем защиты информации является использование упрощенной модели системы.

Рассмотрим случай, когда при статистическом моделировании системы оценивается математическое ожидание дискретной случайной величины (например, вероятность некоторого события).

При оценке математического ожидания дискретной случайной величины оператор модели Ф(y) может быть представлен в виде:

$$\Phi(y) = \begin{cases} C_1, & \text{если} \quad y \in \Omega^{(1)} \\ \dots & \dots \\ C_{L,} & \text{если} \quad y \in \Omega^{(L)} \end{cases},$$

где $C_1,\dots C_L$ – константы, $\Omega_1,\dots\Omega_L$ – непересекающиеся подмножества множества значений y.

Наибольший эффект применение рассматриваемой процедуры дало бы в случае, если границы слоев совпадают с границами множеств $\Omega_1,\dots\Omega_L$, требуя для своего вычисления время, значительно меньшее, чем время, необходимое на воспроизведение исследуемого процесса.

Достигнуть этого можно, используя для определения границ слоев упрощенную модель исследуемого процесса.

Оператор упрощенной модели, обеспечивающий наибольшее уменьшение объема выборки при оценке математического ожидания дискретной случайной величины с требуемой точностью, можно записать в виде:

$$\Phi^*(y) = \begin{cases} C_1, & \text{если} \quad y \in \Omega_*^{(1)} \\ \dots & \dots \\ C_L & \text{если} \quad y \in \Omega_*^{(L)} \end{cases},$$

при определении границ слоев следующим образом:

$$\Omega_1 = \{Y: \Phi^*(Y) = C_1\} = \Omega_*^{(1)}; \dots, \Omega_L = \{Y: \Phi^*(Y) = C_L\} = \Omega_*^{(L)}.$$

Для определения границ множеств $\Omega_*^{(1)}, \dots \Omega_*^{(L)}$ можно использовать кусочно-линейную дискриминантную функцию.

Рассмотрим процедуру построения кусочно-линейной дискриминантной функции, которая может быть использована для определения границ слоев при статистическом моделировании систем защиты информации без больших дополнительных затрат машинного времени.

Выполняя некоторое небольшое количество испытаний исследуемой системы, реализуем розыгрыш случайных величин из различных областей $\Omega_*^{(1)}, \dots \Omega_*^{(L)}$. Примем точки, попавшие в определенную область $\Omega^{(i)}$, за "эталонные" точки для этой области.

Обозначим "эталонные" точки для области $\Omega^{(i)}$, как $P_i^{j_i}(j_i = 1,\dots\bar{n}_i)$, где j_i – номер "эталонной" точки в слое $\Omega^{(i)}$, $\bar{n}_i$ – количество "эталонных" точек в слое $\Omega^{(i)}$.

В процессе "обучения" – получения новых точек в процессе статистического моделирования – значения "эталонных" точек уточняются.

Если вектор y, полученный в очередной реализации. Оказывается, принадлежащим области $\Omega^{(i)}$, то эталонная точка из набора $P_i^{j_i}$, которая ближе всего расположена к y, корректируется по правилу:

$$P_i^{j_i} = (n_i^{j_i} P_i^{j_i} + y)(n_i^{j_i} + 1),$$

где $n_i^{j_i}$ – число предыдущих исправлений этой точки.

Вектор y считается принадлежащим слою $\Omega^{(i)}$, если расстояние $d(y, P_i^{j_i})$ до ближайшей из точек $P_i^{j_i}(j_i = 1,\dots\bar{n}_i)$ меньше, чем расстояние $d(y, P_k^{j_k})$ до ближайшей из точек $P_i^{j_i}(j_i = 1,\dots\bar{n}_i)$ при $k \neq 1$.

Определим расстояния между точкой y и всеми точками $P_i^{j_i}$ из всех слоев $\Omega^{(i)}$ в соответствии с выражением:

$$d\left(y, P_i^{j_i}\right) = \sqrt{\left(y - P_i^{j_i}\right)\left(y - P_i^{j_i}\right)} \quad \left(i = \overline{1, L}; j_i = 1, \dots \bar{n}_i\right),$$

где L –количество слоев, $\bar{n}_i$ – количество "эталонных" точек в слое $\Omega^{(i)}$.

Ближайшей к точке y будет точка $P_i^{j_i}$, в которой достигает максимума функция:

$$g_i^{j_i}(y) = P_i^{j_i} y - P_i^{j_i} P_i^{j_i}/2.$$

Дискриминантная функция $g(y)$ при этом будет определяться выражением:

$$g(y) = \max_{i \in \{1,..L\}} \max_{j_i} g_i^{j_i}.$$

Оператор упрощенной модели в рассматриваемом случае можно записать в виде:

$$\Phi^*(y) = \begin{cases} C_1, , & \text{если} \quad g(y) = g_1^{j_1} \\ \dots & \dots \\ C_L, & \text{если} \quad g(y) = g_L^{j_L} \end{cases},$$

Как видно из приведенного описания процедуры построения планирующей модели, исследователю не требуется при определении границ слоев проводить какую-либо предварительную работу по анализу оператора имитационной модели – определение параметров имитационной модели происходит автоматически на основе полученной в процессе статистического моделирования информации.

Заключение

Рассмотренные в статье процедуры ускоренного статистического моделирования были применены при исследовании различных производственных систем сбора, обработки и защиты информации, позволяя 5-10 раз сократить затраты машинного времени на проведение эксперимента по сравнению со статистическим моделированием по традиционной схеме, что успешно продемонстрировано при решении широкого круга задач, ориентированных на оценку эффективности систем защиты сложных корпоративных вычислительных систем, разнообразных производственных систем [7, 8].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Язов Ю.К., Панфилов А.П.* Методология оценки эффективности защиты информации в информационных системах от несанкционированного доступа: монография. – СПб.: Наукоемкие технологии, 2023. – 258 с.
2. *Бокова О.И., Коробкин Д.И., Кухарев С.А., Попов А.Д.* Разработка имитационной модели системы защиты информации от несанкционированного доступа с использованием программной среды CPN TOOL // Безопасность информационных технологий = IT Security. – 2019. – Т. 26, № 3. – С.80-89.
3. *Данилова О.Т.* Моделирование процесса воздействия вредоносного программного обеспечения на автономную компьютерную систему // Современные методы, средства и технологии защиты информации – 2024: Сборник трудов XV Международной научно-практической конференции имени Олега Борисовича Макаревича (Таганрог, 11–15 сентября 2024 г.); Южный федеральный университет. – Ростов-на-Дону; Таганрог: Изд-во ЮФУ, 2024. – С. 45-52.
4. *Сабинин О.Ю.* Планирование ускоренного статистического моделирования при оценке векторных вероятностных характеристик телекоммуникационных систем // В сборнике: Актуальные проблемы инфотелекоммуникаций в науке и образовании. II Международная научно-техническая и научно-методическая конференция. – 2013. – С. 640-645.

5. *Srinivasan R.* Importance Sampling: Applications in Communications and Detection. – Springer, 2002. – 252 p.
6. *Сабинин О.Ю.* Planning and organization of accelerated statistic modeling for complex industrial technical systems // Izvestiya Rossiiskoi Akademii Nauk. Teoriya i Sistemy Upravleniya. – 1997. – No. 2. – P. 117-123.
7. *Сабинин О.Ю, Серяков П.Ю.* Имитационное моделирование производственно-технических комплексов на основе фрейм-продукционного представления модели // Изв. ГЭТУ: Сб. научн. тр. – СПб., 1998. – Вып. 519.
8. *Помазанов И.Н., Сабинин О.Ю.* Применение событийного моделирования при исследовании систем управления предприятий // Приборы и системы. Управление, контроль, диагностика. – М.: Изд-во «Научтехлитиздат», 2001. – № 7.

УДК 004.056

М.С. Сериккажина

ОБЕСПЕЧЕНИЕ СИСТЕМЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В ОРГАНИЗАЦИЯХ МАЛОГО БИЗНЕСА И РАЗРАБОТКА МЕТОДОВ

В современную эпоху цифровизации и роста числа киберугроз обеспечение защиты информации становится одной из ключевых задач для малого бизнеса. Настоящее исследование посвящено вопросам усовершенствования систем кибербезопасности малых предприятий Казахстана, ориентируясь на мировые технологические достижения. Рассматриваются существующие механизмы защиты данных, анализируются их ограничения, а также оценивается возможность адаптации к условиям малых организаций, зачастую имеющих ограниченные ресурсы на внедрение комплексных решений в области информационной безопасности. Особое внимание уделено современным стратегиям, таким как применение облачных технологий, эффективные методы контроля доступа и актуальные модели предоставления программного обеспечения. В работе также рассматриваются способы минимизации угроз, включая использование многофакторной аутентификации, автоматизированных систем обнаружения атак и технологий мониторинга киберинцидентов. Результаты данного исследования позволяют определить наиболее действенные меры защиты, которые могут повысить уровень кибербезопасности малых предприятий, снизить вероятность утечек данных и привести их деятельность в соответствие с требованиями законодательства. Выводы исследования могут послужить основой для дальнейших научных разработок, а также поспособствовать формированию национальных стандартов в области защиты информации для малого бизнеса.

Ключевые слова: *кибербезопасность, защита информации, малые предприятия, облачные технологии, управление доступом.*

In the context of digitalization and the growing number of cyber threats, ensuring information security is becoming one of the key tasks for small businesses. This study is devoted to improving the cybersecurity systems of small enterprises in Kazakhstan, focusing on global technological advances. The existing data protection mechanisms are reviewed, their limitations are analyzed, and the possibility of adapting to the conditions of small organizations, which often have limited resources to implement integrated information security solutions, is assessed. Special attention is paid to modern strategies such as the use of cloud technologies, effective access control methods and current software delivery models. The paper also discusses ways to minimize threats, including the use of multifactor authentication, automated attack detection systems, and cyber incident monitoring technologies. The results of this study allow us to identify the most effective protection measures that can increase the level of cybersecurity of small enterprises, reduce the likelihood of data leaks and bring their activities in line with legal

requirements. The findings of the study can serve as a basis for further scientific developments, as well as contribute to the formation of national standards in the field of information protection for small businesses.

Keywords: *cybersecurity, information security, small enterprises, cloud technologies, access control.*

Введение

В эпоху бурного развития цифровых технологий и автоматизации бизнес-процессов защита информации играет ключевую роль в обеспечении стабильности и конкурентоспособности компаний. Особенно актуальной эта проблема становится для малых предприятий, которые, обладая ограниченными ресурсами, часто оказываются уязвимыми перед киберугрозами.

Использование облачных сервисов, мобильных устройств и удаленных рабочих мест существенно расширяет потенциальные риски, связанные с утечкой данных и несанкционированным доступом. Если ранее подобные угрозы были характерны преимущественно для крупных корпораций, то сегодня малый бизнес сталкивается с аналогичными вызовами, но зачастую не имеет достаточных инструментов и знаний для их эффективного устранения.

Одной из ключевых проблем является низкий уровень осведомленности предпринимателей о современных киберугрозах и способах защиты. В отличие от крупных компаний, располагающих выделенными отделами информационной безопасности и значительными бюджетами на киберзащиту, малый бизнес в большинстве случаев использует базовые антивирусные решения и слабые пароли, что делает его легкой мишенью для злоумышленников [1].

В Казахстане вопросы защиты информации регламентируются различными нормативными и методическими документами. Однако существующие законодательные требования, как правило, представляют собой общие положения, не содержащие конкретных рекомендаций по построению эффективных систем безопасности. Кроме того, национальные стандарты зачастую не охватывают современные угрозы, что усложняет разработку комплексных решений, адаптированных к потребностям малого бизнеса.

В развитых странах накоплен значительный опыт в области защиты данных и предотвращения внешних атак, включая использование межсетевых экранов, систем обнаружения и предотвращения вторжений (IDS/IPS) и других технологий. Однако растущую угрозу представляют не только хакерские атаки, но и инсайдерские угрозы, связанные с несанкционированным доступом сотрудников или утечкой информации.

Основной целью данного исследования является анализ актуальных угроз информационной безопасности в Казахстане, выявление наиболее эффективных методов защиты и оценка их применимости для малых пред-

приятий. Работа охватывает как технические, так и организационные аспекты кибербезопасности, а также включает сравнительный анализ отечественного опыта с лучшими мировыми практиками.

Проблема информационной безопасности в малом бизнесе становится все более значимой в условиях стремительного роста числа кибератак. Согласно статистическим данным, на предприятия малого и среднего бизнеса приходится более 43% всех киберинцидентов. Среди ключевых факторов, препятствующих эффективной защите данных, можно выделить нехватку осведомленности среди предпринимателей, дефицит квалифицированных специалистов и ограниченные финансовые возможности для внедрения современных решений [2].

Кроме того, на малые компании накладываются все более строгие требования в области защиты персональных данных и корпоративной информации. В Казахстане действует Закон «О персональных данных и их защите», а также ряд международных стандартов, таких как европейский GDPR, регулирующий обработку данных в компаниях, сотрудничающих с зарубежными партнерами.

С развитием технологий малый бизнес получает доступ к новым инструментам защиты данных, среди которых:

- Интеграция облачных сервисов с усиленной системой безопасности (Google Workspace, Microsoft 365).
- Использование искусственного интеллекта и автоматизированных систем мониторинга киберугроз.
- Внедрение многофакторной аутентификации и строгих политик управления доступом.
- Повышение цифровой грамотности сотрудников через обучение основам кибербезопасности.
- Развитие государственных программ поддержки малого бизнеса в области защиты информации.

Таким образом, вопросы информационной безопасности в малых компаниях требуют комплексного подхода, сочетающего современные технологии, обучающие программы и государственное регулирование.

Методология

Киберугрозы становятся все более сложными и разнообразными, а малые предприятия, обладая ограниченными ресурсами, нередко оказываются наиболее уязвимыми перед атаками. В рамках исследования были рассмотрены наиболее распространенные угрозы, с которыми сталкивается малый бизнес, а также методы защиты, применяемые как в Казахстане, так и за рубежом. Основные угрозы для малого бизнеса:

Фишинг – попытки злоумышленников получить учетные данные сотрудников и клиентов через поддельные письма и веб-сайты. По данным исследований, около 90% всех успешных кибератак начинается именно с фишинга.

Вредоносное ПО (вирусы, программы-вымогатели, трояны) – блокировка данных компании с требованием выкупа. В последние годы участились атаки программ-вымогателей, зашифровывающих данные организаций.

DDoS-атаки – перегрузка серверов компании, ведущая к временной недоступности сервисов, финансовым потерям и снижению доверия клиентов.

Социальная инженерия – манипулирование сотрудниками с целью получения доступа к конфиденциальной информации.

Атаки на слабозащищенные системы – использование устаревшего ПО, ненадежных паролей и отсутствия многофакторной аутентификации.

По данным аналитических агентств, 43% всех кибератак направлены на малый бизнес. Однако 60% малых предприятий не имеют планов реагирования, а 29% компаний, столкнувшись с серьезной атакой, прекращают деятельность.

Например, в 2021 году небольшая бухгалтерская фирма в Казахстане стала жертвой программы-вымогателя, зашифровавшего клиентские данные. Отсутствие резервных копий вынудило компанию выплатить выкуп.

В 2022 году казахстанский интернет-магазин подвергся DDoS-атаке во время распродажи, что привело к длительному простою и финансовым потерям [3].

Сравнивая мировую и Казахстанскую информационную безопасность, можно заметить ряд различий. В Казахстане малый бизнес сталкивается с рядом проблем в сфере информационной безопасности.

Ограниченные бюджеты на защитные меры, отсутствие антивирусных программ и аудитов безопасности.

Нехватка квалифицированных специалистов, так как малый бизнес не может позволить себе содержать ИТ-отдел.

Недостаток строгих законодательных требований. В отличие от США и ЕС, в Казахстане нет обязательных стандартов защиты для всех предприятий.

Мировые практики:

США: Федеральная торговая комиссия (FTC) обязывает компании соблюдать стандарты защиты данных.

Европа: В рамках GDPR малый бизнес должен обучать сотрудников кибербезопасности и внедрять защитные механизмы.

В Казахстане подобные практики внедряются преимущественно в крупных компаниях, тогда как малый бизнес редко уделяет внимание профилактическим мерам. Однако современные технологии позволяют минимизировать затраты на защиту, например, за счет использования облачных решений по модели SaaS (Security as a Service).

Пример: В одной из немецких компаний после внедрения системы автоматического обнаружения аномалий количество инцидентов сократилось на 45%.

Аудиты безопасности – регулярные проверки защиты данных помогают выявлять слабые места.

В Германии проводится регулярный аудит безопасности для всех предприятий, включая малый бизнес, что позволяет выявлять слабые места и предотвращать атаки.

К основным формам системы информационной безопасности относятся:

1) правовой;
2) организационные;
3) аппаратно-программный.

Правовые методы являются основой обеспечения ИБ, поскольку субъекты обязаны выполнять все требования законодательства.

Одним из важнейших аспектов законодательного уровня является своевременное пересмотр законов в соответствии с процессами развития информационных технологий.

В случае несвоевременного пересмотра законов и иных нормативных актов произошедший процесс влечет снижение уровня информационной безопасности. Например, Агентство РК» Агентство по информатизации и связи «инициировало внесение дополнений и изменений в Закон РК» Об информатизации", принятый в 1994 году, так как при разработке и принятии закона невозможно предсказать стремительное развитие информационных технологий, которое произойдет в ближайшие 10 лет[4].

Организационные методы также являются фундаментом информационной безопасности и включают:

Разработка политики безопасности – создание четких правил работы с конфиденциальной информацией, определение уровней доступа, ведение логов действий сотрудников.

Внедрение многофакторной аутентификации (MFA) – использование паролей в сочетании с биометрией, токенами или SMS-кодами для защиты учетных записей.

Проведение регулярных тренингов – обучение сотрудников принципам кибербезопасности, таким как защита от фишинга, безопасное использование паролей и выявление подозрительных действий.

Назначение ответственного за ИБ – в малых компаниях редко бывает отдельный отдел безопасности, но даже один обученный сотрудник может значительно повысить уровень защиты.

Создание плана реагирования на инциденты – алгоритмы действий в случае утечки данных, атак или взломов, включая уведомление клиентов и партнеров, восстановление данных и устранение уязвимостей.

Пример: В одной из малых IT-компаний Казахстана после внедрения политик безопасности и регулярных тренингов количество успешных фишинговых атак сократилось на 60% в течение года.

К числу эффективных технических решений относятся:

Использование антивирусных программ и межсетевых экранов – фильтрация входящего трафика, блокировка вредоносных программ и защита от вирусов.

Шифрование данных – использование алгоритмов AES-256 или RSA для защиты данных в хранилищах и при передаче через сеть.

Создание резервных копий – автоматическое резервное копирование критически важных данных с хранением копий в облаке или на внешних серверах.

Мониторинг активности в сети – внедрение SIEM-систем (Security Information and Event Management) для анализа логов и обнаружения подозрительных действий.

Обновление программного обеспечения – устранение уязвимостей через своевременное обновление ОС, приложений и системных компонентов.

Пример: В одной из торговых компаний после внедрения системы шифрования данных утечки информации о клиентах сократились на 80%.

Результаты и обсуждение

Для оценки эффективности предложенных мер были проведены исследования в нескольких малых предприятиях Казахстана. В ходе эксперимента использовались:

- Политика безопасности, регламентирующая порядок работы с конфиденциальной информацией.
- Тренинги для сотрудников, направленные на повышение осведомленности о киберугрозах.
- Антивирусные решения и межсетевые экраны для фильтрации вредоносного трафика.
- Внедрение многофакторной аутентификации для защиты учетных записей.
- Регулярное резервное копирование данных для предотвращения потерь.

Результаты показали, что внедрение даже базовых мер привело к снижению киберинцидентов: в Казахстане количество атак сократилось на 30%.

Дополнительно было выявлено, что компании, уделяющие больше внимания обучению персонала, демонстрируют более высокий уровень защиты. Например, предприятия, проводившие тренинги по фишингу, снизили риск утечек данных на 40%.

Применение организационных и технических мер показало различную эффективность в зависимости от уровня зрелости ИБ-инфраструктуры компании. В странах ЕС внедрение стандартов безопасности и сертификации играет ключевую роль в снижении угроз. Например:

В Германии обязательные аудиты безопасности и сертификация по стандарту ISO/IEC 27001 повышают уровень защиты данных.

В Великобритании действуют программы по повышению киберграмотности среди сотрудников малых предприятий.

В США активное использование искусственного интеллекта и автоматизированных решений позволяет в реальном времени выявлять угрозы и предотвращать атаки.

В Казахстане пока не разработаны единые обязательные стандарты кибербезопасности для малого бизнеса, что усложняет ситуацию. Однако компании, внедряющие международные практики, демонстрируют рост устойчивости к угрозам. Например, организация, внедрившая систему мониторинга сети и регулярные обновления ПО, за год сократила количество атак на 35%.

Также стоит отметить, что финансирование играет ключевую роль. В ЕС и США существуют государственные программы поддержки малого бизнеса в области кибербезопасности, в то время как в Казахстане подобные инициативы только начинают развиваться.

На основе проведенного анализа можно выделить ключевые рекомендации для повышения уровня информационной безопасности в малом бизнесе Казахстана:

- Разработка национальных стандартов кибербезопасности – обязательные требования к защите данных, аналогичные GDPR в Европе.
- Создание государственных программ поддержки – субсидии и гранты для малого бизнеса на внедрение ИБ-решений.
- Популяризация кибербезопасности – образовательные программы, онлайн-курсы и тренинги для предпринимателей и сотрудников.
- Развитие партнерств с крупными компаниями – обмен опытом и использование передовых технологий крупных предприятий.

Внедрение этих рекомендаций позволит значительно повысить уровень защиты данных и снизить количество атак на малые предприятия, делая их более устойчивыми к современным угрозам.

Для защиты данных малый бизнес должен создать надежную ИБ-инфраструктуру, включающую следующие ключевые компоненты:

- Система управления доступом (Identity and Access Management, IAM) играет ключевую роль в предотвращении несанкционированного доступа к корпоративным ресурсам. Она включает:
- Многофакторную аутентификацию (MFA) – требование ввода нескольких факторов при входе (пароль + код из SMS, биометрия и т.д.).
- Ролевое управление доступом (RBAC) – назначение прав пользователям в зависимости от их **должностных обязанностей.**
- **Журналирование входов и событий** – ведение логов, позволяющее отслеживать подозрительную активность.

Пример внедрения: Компании, использующие IAM-системы, снижают риск утечки учетных данных на 60%.

Резервное копирование критически важно для восстановления данных после кибератак или технических сбоев. Лучшие практики включают:

- Автоматическое создание резервных копий – настройка регулярных бэкапов.
- Хранение копий в географически распределенных центрах данных – защита от локальных катастроф.
- Шифрование бэкапов – защита данных от компрометации.

Пример внедрения: По данным IBM, компании с облачным резервным копированием восстанавливают работу после атак на 50% быстрее.

Система обнаружения и предотвращения вторжений (IDS/IPS)

- IDS/IPS анализируют сетевой трафик и блокируют подозрительные активности. Их возможности включают:
- Выявление аномалий – обнаружение необычного поведения в сети.
- Автоматическое реагиро**вание** – блокировка атак в реальном времени.
- Интеграция с SIEM-системами – централизованное управление инцидентами безопасности.

Пример внедрения: В малом бизнесе IDS-системы предотвращают до 80% потенциальных угроз.

VPN для безопасного удаленного доступа

- Виртуальные частные сети (VPN) обеспечивают защищенный доступ сотрудников к корпоративным ресурсам из любых мест. Основные преимущества:
- Шифрование трафика – защита данных от перехвата.
- Аутентификация пользователей – предотвращение доступа злоумышленников.
- Защита конфиденциальности – скрытие IP-адресов сотрудников.

Пример внедрения: VPN значительно снижает риск атак на удаленных сотрудников, что особенно важно при работе из дома.

Антивирусные решения и межсетевые экраны

- Антивирусное ПО защищает от вредоносных программ и фишинговых атак.
- Межсетевые экраны (firewalls) фильтруют входящий и исходящий трафик, предотвращая несанкционированные подключения.

Пример внедрения: По данным Microsoft, использование современных антивирусов снижает вероятность заражения на 90%.

Системы логирования и мониторинга позволяют анализировать инциденты безопасности и своевременно реагировать на угрозы. Основные компоненты:

- Централизованные логи – сбор данных со всех устройств и сервисов.
- Автоматический анализ событий – выявление подозрительной активности.
- Оповещения в режиме реального времени – быстрая реакция на инциденты.

Пример внедрения: Компании, использующие логирование, выявляют 95% угроз до того, как они приведут к серьезным последствиям.

ИИ и машинное обучение позволяют анализировать сетевой трафик и выявлять аномалии, предотвращая атаки до их осуществления. Основные направления использования:

- Анализ поведения пользователей – системы отслеживают, какие действия характерны для каждого сотрудника, и сигнализируют о подозрительных изменениях.
- Выявление фишинговых атак – алгоритмы анализируют содержание электронных писем и сайтов, выявляя поддельные ресурсы.
- Обнаружение вредоносного кода – машинное обучение помогает находить вредоносные программы, даже если они ранее не встречались.
- Прогнозирование угроз – на основе анализа больших данных ИИ определяет возможные уязвимости и рекомендует меры защиты.

Пример: Компании, такие как Darktrace и IBM Watson Security, используют ИИ для мониторинга сетей и автоматического реагирования на инциденты.

Блокчейн – это технология децентрализованного хранения данных, которая обеспечивает их неизменность и защиту от фальсификации. Основные применения:

- Безопасное хранение логов – верификация действий сотрудников и контроль изменений данных.

- Защита финансовых транзакций – использование смарт-контрактов для автоматизированного выполнения платежей.
- Децентрализованные системы аутентификации – пользователи могут подтверждать свою личность без хранения паролей в централизованных базах данных.

Пример: Блокчейн-система Guardtime используется правительством Эстонии для защиты государственных данных.

Традиционные пароли могут быть скомпрометированы, поэтому биометрия становится более надежной альтернативой. Популярные технологии:

- Распознавание лиц – например, Face ID в iPhone.
- Сканирование отпечатков пальцев – используется в ноутбуках и корпоративных системах доступа.
- Распознавание голоса и сетчатки глаза – повышенная защита для банковских и государственных систем.

Пример: Банки, такие как HSBC и Citibank, внедрили голосовую биометрию для идентификации клиентов.

Надежная ИБ-инфраструктура в малом бизнесе должна включать комплекс технических решений, направленных на предотвращение атак, защиту данных и мониторинг безопасности. Инвестирование в базовые технологии позволяет значительно снизить риски и защитить информацию компании.

Заключение

Обеспечение безопасности информации в малом бизнесе требует целостного подхода, охватывающего как организационные, так и технические и правовые аспекты. Анализ текущей ситуации в Казахстане демонстрирует, что предприятия небольшого размера остаются подвержены различным угрозам из-за ограниченного финансирования, нехватки квалифицированных кадров и недостаточно развитого законодательства в области кибербезопасности.

На данный момент отсутствуют четкие методические указания, регламентирующие приведение информационных систем организаций в соответствие с законодательными нормами. В рамках данного исследования были сформулированы рекомендации, направленные на совершенствование существующих механизмов защиты, их внедрение и адаптацию под потребности предприятий.

Согласно проведенному анализу, построение системы информационной безопасности следует рассматривать как масштабный проект, охватывающий все уровни и аспекты защиты данных. В его основе должны лежать требования национального законодательства, отраслевых нормативов и международных стандартов, применимых к конкретному бизнесу. Важным элементом обеспечения защиты информации является не только внедрение технических решений, но и реализация организационных мероприятий, включая разработку внутренних регламентов и инструкций, регулирующих бизнес-

процессы и управление данными. Без четко сформулированных стандартов невозможно достичь высокого уровня защищенности информационных активов компании.

Опыт развитых государств, таких как США и страны Европейского Союза, показывает, что использование обязательных стандартов (например, GDPR), применение автоматизированных систем мониторинга и проведение образовательных программ для сотрудников позволяет существенно снизить риски кибератак. По данным исследования Verizon Data Breach Investigations Report, свыше 60% инцидентов безопасности в малом бизнесе могли бы быть предотвращены посредством базовых мер защиты, таких как многофакторная аутентификация и повышение осведомленности персонала [5].

Перспективные направления дальнейшего развития данной сферы включают:

Применение технологий искусственного интеллекта для автоматического выявления и предотвращения угроз;

Разработку государственных инициатив по поддержке малого бизнеса в вопросах кибербезопасности;

Совершенствование нормативно-правовой базы Казахстана, включая регулирование защиты персональных данных и противодействие цифровым угрозам.

Таким образом, внедрение мирового опыта и развитие отечественных программ кибербезопасности позволит значительно повысить уровень защиты данных малых предприятий в Казахстане, минимизируя финансовые и репутационные риски, связанные с киберинцидентами.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Путято М.М., Макарян А.С.* Кибербезопасность как неотъемлемый атрибут многоуровневого защищенно-го киберпространства // Прикаспийский журнал: управление и высокие технологии. – 2020. – № 3 (51). – С. 94-102.
2. *Васильков А.В., Васильков А.А., Васильков И.А.* Информационные системы и их безопасность: учебное пособие. – М.: Форум, 2013. – 153 с.
3. https://liter.kz/khaker-treboval-ot-kazakhstanskoi-kompanii-vykup-v-kriptovaliute-za-rasshifrovku-bukhgalterskoi-programmy-1649754573/.
4. Қазахстан Республикасының 11.01.2007 № 217-III «Ақпараттандыру» туралы заңы.
5. *Уткин В.Б., Балдин К.В.* Информационные системы в экономике: учебник для студентов высших учебных заведений. – М.: ИЦ Академия, 2012. – 191 с.

УДК 004.89

С.С. Скрыльников

ИСПОЛЬЗОВАНИЕ СЕМАНТИЧЕСКОГО ПОИСКА В КАЧЕСТВЕ МОДУЛЯ DLP-СИСТЕМЫ

Современные системы предотвращения утечек данных (DLP) используют анализ текстового содержимого, который является критически важным модулем для обнаружения и предотвращения несанкционированной передачи конфиденциальной информации. Изначально, для решения этой задачи применяются методы морфологического и синтаксического анализа, которые, несмотря на свою распространенность, имеют ряд существенных ограничений. К ним относятся высокая зависимость от грамматической структуры языка, трудности в обработке полисемии и идиоматических выражений, а также ограниченная способность выявлять скрытые или подразумеваемые смыслы. Эти недостатки делают традиционные подходы уязвимыми к попыткам обхода путем использования синонимов, перефразирования или контекстуальных замен. В данной работе проводится обзор современных DLP-систем и анализируются применяемые в них подходы к лингвистическому анализу, выделяются их ключевые преимущества и недостатки в контексте обнаружения утечек данных. В качестве перспективной альтернативы, способной преодолеть ограничения традиционных методов, предлагается использование семантического поиска. Этот подход основан на современных моделях представления значений слов в числовом векторном виде и контекстного понимания текста, что позволяет системам DLP учитывать смысловую близость терминов и обнаруживать утечки информации даже при использовании лексических вариаций и словоформ. В работе подробно рассматриваются потенциальные преимущества семантического поиска перед традиционными методами анализа текста в DLP-системах, включая повышенную точность обнаружения и устойчивость к попыткам обхода. Наряду с этим, обсуждаются возможные трудности и вызовы, связанные с внедрением семантического поиска, такие как вычислительная сложность, необходимость в больших объемах размеченных данных для обучения моделей и интерпретация результатов семантического анализа в контексте политик безопасности данных. Предлагаются направления для дальнейших исследований, направленных на эффективную интеграцию семантического поиска в DLP-системы для повышения их надежности и эффективности в борьбе с утечками конфиденциальной информации.

Ключевые слова: *предотвращение утечек данных, DLP-системы, лингвистический анализ, семантический поиск, информационная безопасность, анализ текста.*

Modern Data Loss Prevention (DLP) systems implement textual content analysis being a critical component for detecting and preventing the unauthorized transmission of confidential information. Traditionally, this task has been addressed through morphological and syntactic analysis techniques which, despite their widespread use, exhibit several

significant limitations. These include a strong dependency on grammatical structures, difficulties in handling polysemous words and idiomatic expressions, and limited capability to identify implicit or concealed meanings. Such shortcomings render traditional approaches vulnerable to evasion through synonym substitution, paraphrasing, or contextual rewording. This study provides an overview of current DLP systems and examines the linguistic analysis methods employed within them, highlighting their key strengths and weaknesses in the context of data leakage detection. As a promising alternative capable of overcoming the limitations of traditional approaches, the paper proposes the use of semantic search. This method leverages advanced word embeddings and contextual text understanding, enabling DLP systems to capture semantic similarities and detect information leaks even with use of lexical variations. The paper explores the potential advantages of semantic search over conventional text analysis techniques in DLP systems, including enhanced detection accuracy and increased resilience to circumvention attempts. Additionally, it discusses the challenges associated with implementing semantic search, such as computational complexity, the need for large volumes of annotated training data, and the interpretation of semantic analysis results within the framework of data security policies. Finally, the study outlines directions for future research aimed at the effective integration of semantic search into DLP systems to improve their reliability and efficiency in countering confidential data leaks.

Keywords: *data loss prevention, DLP systems, linguistic analysis, semantic search, information security, text analysis.*

Введение

В современном цифровом мире организации сталкиваются с беспрецедентными объемами данных, циркулирующих между различными системами, облачными сервисами и конечными устройствами. Защита конфиденциальной информации от несанкционированного доступа, использования или передачи стала критически важной задачей. Системы предотвращения утечек данных (DLP) играют ключевую роль в решении этой задачи, помогая организациям контролировать и защищать свои ценные активы данных. Эти системы помогают соблюдать нормативные требования, такие как общий регламент по защите данных (GDPR), защита медицинской информации (HIPAA) и безопасность финансовых транзакций (PCI DSS). DLP-системы осуществляют мониторинг данных в различных состояниях: при хранении, передаче и использовании.

Растущая сложность IT-инфраструктур, включая облачные решения, локальные системы и разнообразие конечных устройств, а также увеличение числа сотрудников, работающих удаленно, значительно усложняют задачу обеспечения безопасности данных. В этих условиях надежные DLP-решения становятся необходимостью. Утечки данных могут привести к серьезным финансовым потерям и непоправимому ущербу для репутации компании, что делает инвестиции в эффективные DLP-системы оправданными.

Традиционные DLP-системы часто полагаются на методы лингвистического анализа, такие как поиск по ключевым словам и регулярным выражениям [1]. Однако, эти методы имеют свои ограничения, особенно при работе с неструктурированными данными и понимании контекста информации. Подход, основанный на поиске точных совпадений ключевых слов, может приводить к ложным срабатываниям, когда данные содержат те же слова, что и конфиденциальные, и к ложным пропускам, когда закрытая информация выражена другими словами или фразами [2]. Регулярные выражения, хотя и мощны для поиска определенных паттернов, могут быть сложны в настройке и поддержке, а также могут не учитывать контекст [3].

В качестве потенциального решения для преодоления этих ограничений рассматривается интеграция семантического поиска в DLP-системы. Семантический поиск использует технологии обработки естественного языка (NLP) и машинного обучения для понимания смысла и контекста данных [4]. Такой подход позволяет идентифицировать конфиденциальную информацию не только по наличию определенных слов, но и по ее смысловому содержанию. Интеграция семантического поиска в DLP влечёт повышение точности обнаружения утечек и снижение количества ложных срабатываний за счет более глубокого понимания контекста данных, вместо поиска конкретных текстовых совпадений.

Лингвистический анализ в современных DLP-системах

Современные DLP-системы используют различные методы для анализа контента, включая поиск по ключевым словам, использование регулярных выражений и более продвинутые техники, основывающиеся на морфологии и комбинации первых двух [5]. Лингвистический анализ в этих системах направлен на изучение содержания файлов, игнорируя такие метаданные, как имена файлов. К основным техникам относятся морфологический анализ, который позволяет находить все возможные формы слова, и семантический анализ, который заключается в поиске ключевой информации и оценке контекста ее использования, использующие словари в качестве основы [6].

Поиск по ключевым словам является одним из наиболее распространенных методов, используемых в DLP-системах. Он заключается в идентификации заранее определенных слов или фраз в анализируемом контенте. Этот метод эффективен для обнаружения структурированных данных и известных терминов. Например, DLP-система может быть настроена на поиск таких слов, как «конфиденциально», «коммерческая тайна» или названий внутренних проектов [7]. Регулярные выражения представляют собой еще один широко используемый инструмент, позволяющий определять шабло-

ны для поиска определенных последовательностей символов. Они особенно полезны для идентификации структурированных данных, таких как номера кредитных карт или номера социального страхования, которые имеют четко определенный формат. Простота реализации и эффективность при поиске точных совпадений известных конфиденциальных терминов, особенно в структурированных данных, делают поиск по ключевым словам важным начальным уровнем защиты в DLP-системах [8]. Регулярные выражения обеспечивают гибкость в определении сложных шаблонов, что позволяет выявлять структурированные конфиденциальные данные, такие как идентификационные номера или специфические форматы данных.

Несмотря на свою полезность, традиционные методы лингвистического анализа имеют ряд ограничений. Поиск по ключевым словам не учитывает контекст и может быть легко обойден путем использования синонимов или перефразирования. Например, документ, содержащий описание конфиденциальной информации, но не включающий заданные ключевые слова, может быть пропущен DLP-системой. Регулярные выражения, хотя и мощны для работы со структурированными данными, плохо справляются с неструктурированной информацией, где конфиденциальные данные могут быть представлены на естественном языке без строгого соблюдения определенного формата [9]. Оба метода могут генерировать большое количество ложных срабатываний, в случаях, где контент случайно содержит те же ключевые слова или соответствует заданным шаблонам, что приводит к увеличению труда специалистов по безопасности и снижает эффективность DLP-системы [10]. Ограниченность поиска по ключевым словам проявляется в его неспособности оценивать семантическое значение, что делает его уязвимым к вариациям в формулировках конфиденциальной информации. Сложность создания точных регулярных выражений и их ограниченная способность интерпретировать контекст делают их менее эффективными для выявления конфиденциальной информации в неструктурированном текстовом контенте. Высокий уровень ложных срабатываний, генерируемых DLP-системами на основе ключевых слов и регулярных выражений, может перегружать команды безопасности, отвлекая их от реальных угроз безопасности [11]. Тем не менее, из-за именно данные методы являются наиболее используемыми из-за простоты реализации и в текущий момент используются в большом количестве DLP-систем (табл. 1).

Таблица 1

Применение текстового анализа в современных DLP-системах

DLP-система	Методы текстового анализа
SearchInform	Поиск по ключевым словам/фразам. Использование регулярных выражений. Технология «Поиск похожих» с учётом морфологии Tsearch2. Полнотекстовый поиск Elastic Search.
Solar Dozor	Контентная фильтрация (на основе политик безопасности). Анализ архива электронной почты.
DeviceLock DLP Suite	Метод цифровых отпечатков для классификации файлов Контентный анализ для предотвращения утечек и блокировки пересылки.
Zecurion DLP	Гибридный анализ. Обучаемая технология поиска SmartID. Метод опорных векторов SVM.
Кибер Протего	Индексация данных, использование метаданных, включая журналы нажатия клавиш - кейлоггер. Классификация типов файлов.
Safecopy	Технология маркировки документов.
StaffCop	Контентные фильтры (гибкие конфигурации).
Дозор-Джет	Разграничение доступа классов файлов.
InfoWatch	Классификация типов файлов с использованием распознавания символов (OCR). Применение политик безопасности к классам файлов.
Стахановец	Не имеет специфических методов текстового анализа. Основан на поведенческом анализе, контроле действий, биометрии.
Falcongaze	

Использование семантического поиска

Семантический поиск представляет собой метод поиска информации, целью которого является понимание намерения пользователя и смысла контента, а не простое сопоставление ключевых слов. Он используется в области обработки естественного языка (NLP) для решения таких задач как классификация текста, распознавание сущностей и контекстное понимание. В отличие от традиционного поиска по ключевым словам, который возвра-

щает результаты на основе наличия определенных терминов без учета их значения или контекста, семантический поиск анализирует отношения между словами и общий контекст, чтобы интерпретировать смысл. Он способен распознавать синонимы, связанные понятия и намерение, стоящее за запросом или контентом.

В основе семантического поиска лежат принципы обработки естественного языка (NLP) и машинного обучения. NLP позволяет машинам обрабатывать и понимать естественный язык. Алгоритмы машинного обучения обучаются на больших наборах данных для выявления закономерностей и связей в тексте, что повышает точность семантического анализа. Такие методы, как анализ тональности, извлечение отношений и разрешение многозначности слов, способствуют лучшему пониманию контекста. NLP предоставляет основные инструменты для семантического поиска, позволяя анализировать структуру текста, его значение и связи между сущностями. Машинное обучение расширяет возможности семантического поиска, позволяя системе учиться на данных, адаптироваться к новым закономерностям и улучшать своё понимание текста [12, 13].

В контексте DLP-систем семантический поиск может быть использован для идентификации конфиденциальной информации на основе ее смысла, а не только ключевых слов. DLP-система, оснащенная семантическим поиском, способна анализировать содержание документов, электронных писем и других источников данных, чтобы понять контекст использования закрытой информации. Например, она может определить документ, обсуждающий финансовые результаты, как конфиденциальный, даже если в нем отсутствуют конкретные ключевые слова, такие как "конфиденциально" или "секретно". Семантический поиск также позволяет различать номер ссылки и номер социального страхования на основе окружающего текста и контекста [14]. Понимание смыслового содержания позволяет DLP-системам выявлять конфиденциальную информацию даже при ее перефразировании или косвенном упоминании, что устраняет недостатки обнаружения на основе ключевых слов. В данный момент семантический поиск широко применяется в интеллектуальных чат-ботах, рекомендательных и поисковых системах, для определения наиболее релевантных и контекстно подходящих результатов.

Преимущества семантического поиска в DLP-системах

Интеграция семантического поиска в DLP-системы предоставляет ряд значительных преимуществ по сравнению с традиционными методами [15]. Одним из ключевых преимуществ является повышение точности обнаружения конфиденциальных данных, особенно в неструктурированных форматах. Семантический поиск, основанный на NLP, отлично справляется с анализом неструктурированных данных, таких как электронные письма, документы и презентации, где конфиденциальная информация часто встроена в

естественный язык [16]. Он способен идентифицировать закрытые концепции и сущности, даже если точные ключевые слова отсутствуют. Способность семантического поиска понимать смысл неструктурированных данных значительно повышает точность DLP в выявлении конфиденциальной информации, которую традиционные методы часто пропускают [17].

Еще одним важным преимуществом является потенциальное значительное снижение количества ложных срабатываний. Благодаря пониманию контекста семантический поиск может различать закрытое и общее использование одинаковых терминов, что приводит к уменьшению количества ложных тревог, что снижает уровень "шума" для команд безопасности, позволяя им сосредоточиться на реальных угрозах. Тем самым повышая эффективность работы DLP-систем. Уменьшение усталости от оповещений позволяет сотрудникам безопасности более оперативно реагировать на фактические инциденты [18].

Семантический поиск также расширяет возможности DLP-систем по пониманию контекста данных и действий пользователей, способствуя поведенческой аналитике, позволяя понимать смысл коммуникаций и действий пользователей. Семантический анализ позволяет DLP выходить за рамки простого выявления конфиденциальных данных и интерпретировать собственное использование, предоставляя ценный контекст для оценки рисков и реагирования на инциденты. Сравнение реализаций и характеристик методов лингвистического анализа представлено в табл. 2.

Таблица 2

Сравнение традиционного лингвистического анализа и семантического поиска в DLP-системах

Характеристика	Ключевые слова	Регулярные выражения	Семантический поиск
Основа анализа	Ключевые слова	Шаблоны	Смысл и контекст
Обработка неструктурированных данных	Плохо	Ограниченно	Хорошо
Понимание контекста	Отсутствует	Ограниченно	Высокое
Точность	Низкая	Средняя	Высокая
Уровень ложных срабатываний	Высокий	Средний	Низкий
Сложность внедрения	Низкая	Средняя	Высокая
Зависимость от предопределенных правил	Высокая	Высокая	Низкая

Особенности внедрения семантического поиска в DLP-системы

Внедрение семантического поиска в DLP-системы сопряжено с определенными практическими соображениями и сложностями. Одним из ключевых аспектов являются требования к вычислительным ресурсам. Семантический анализ, особенно с использованием передовых моделей NLP и машинного обучения, может быть весьма ресурсоемким, требуя значительной вычислительной мощности и объема памяти [19]. Анализ больших объемов данных в режиме реального времени может создавать нагрузку на IT-инфраструктуру. Внедрение семантического поиска в DLP-системы может потребовать значительных вычислительных ресурсов, что может привести к необходимости обновления или оптимизации инфраструктуры [20].

Еще одной сложностью является интеграция модулей семантического поиска с существующими архитектурами DLP и источниками данных. Интеграция модуля семантического поиска с уже развернутой DLP-системой может потребовать существенных архитектурных изменений и тщательного планирования для обеспечения совместимости и эффективного потока данных. Подключение и анализ данных из различных источников, таких как облачные сервисы, локальные системы и конечные устройства, может быть непростой задачей. Бесшовная интеграция возможностей семантического поиска с существующей инфраструктурой DLP и разнообразными источниками данных представляет собой значительную техническую проблему, требующую тщательного рассмотрения.

Наконец, для разработки и поддержки модуля семантического поиска требуется наличие специализированных знаний в области NLP, машинного обучения и науки о данных. Организациям может потребоваться нанимать или обучать сотрудников, обладающих этими навыками. Успешное внедрение и дальнейшее обслуживание семантического поиска в DLP-системах требует команды со специальными знаниями в области передового анализа данных и методов машинного обучения.

Заключение

Семантический поиск представляет собой многообещающий подход к значительному улучшению возможностей DLP-систем за счет повышения точности обнаружения. Его способность понимать смысл и контекст данных решает многие ограничения традиционных методов, основанных на ключевых словах и регулярных выражениях.

В будущем, с ростом объемов неструктурированных данных и увеличением популярности облачных решений, вероятно, будет наблюдаться более широкое внедрение семантического поиска и NLP в DLP-системах. Интеграция с искусственным интеллектом и машинным обучением будет способствовать дальнейшему повышению интеллектуальности и адаптивности DLP-систем. Увеличение использования облачных сервисов и распростра-

нение неструктурированных данных сделают семантический поиск все более важным компонентом эффективных стратегий DLP. Интеграция ИИ и МО с семантическим поиском приведет к созданию более совершенных DLP-систем, способных проактивно выявлять и реагировать на риски утечки данных с большей точностью и автоматизацией.

Несмотря на существующие сложности с внедрением, преимущества семантического поиска в плане точности и снижения уровня "шума" делают его ценным дополнением к ландшафту DLP, открывая путь к созданию более интеллектуальных и эффективных стратегий предотвращения утечек данных, выходящих за рамки простого сопоставления шаблонов и обеспечивающих истинное понимание контента.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Гречанная А.Ю., Тастенов А.Д.* DLP-системы и их роль в защите от утечек конфиденциальной информации // Наука и техника Казахстана. – 2015. – № 3-4. – URL: https://cyberleninka.ru/article/n/dlp-sistemy-i-ih-rol-v-zaschite-ot-utechek-konfidentsialnoy-informatsii (дата обращения: 06.04.2025).
2. *Чуб В.С., Айдинян А.Р.* Методика повышения эффективности DLP-систем для защиты конфиденциальной информации // Инновационная наука. – 2017. – № 4-3. – URL: https://cyberleninka.ru/article/n/metodika-povysheniya-effektivnosti-dlp-sistem-dlya-zaschity-konfidentsialnoy-informatsii (дата обращения: 06.04.2025).
3. *Страхов А.А., Дубинина Н.М.* Об утечке данных и DLP-системах // Криминологический журнал. – 2022. – № 4. – URL: https://cyberleninka.ru/article/n/ob-utechke-dannyh-i-dlp-sistemah (дата обращения: 06.04.2025).
4. *Артюшкина Е.С., Скакун О.О., Гузь А.Р.* Использование искусственного интеллекта в DLP-системах // Прикладные экономические исследования. – 2023. – № 2. – URL: https://cyberleninka.ru/article/n/ispolzovanie-iskusstvennogo-intellekta-v-dlp-sistemah (дата обращения: 06.04.2025).
5. *Королев В.В.* Использование методов анализа контента в DLP системах // Проблемы науки. – 2016. – № 10 (11). – URL: https://cyberleninka.ru/article/n/ispolzovanie-metodov-analiza-kontenta-v-dlp-sistemah (дата обращения: 06.04.2025).
6. *Кумунжиев К.В., Зверев И.Н.* Метод повышения эффективности DLP-системы при семантическом анализе и категоризации информации // Современные проблемы науки и образования. – 2014. – № 5. – URL: https://science-education.ru/ru/article/view?id=14741 (дата обращения: 02.03.2025).
7. Keyword Rules – Skyhigh Security [Электронный ресурс]. – URL: https://success.skyhighsecurity.com/Skyhigh_Data_Loss_Prevention/Sanctioned_DLP_Policies_and_Rules/Sanctioned_DLP_Policies_Rules_and_Rule_Groups/Keyword_Rules (дата обращения: 05.04.2025).
8. *Зверев И.Н.* Сравнительный обзор методов категоризации, применяемых в DLP-системах // Актуальные вопросы современной науки. – 2014. – № 35. – URL: https://cyberleninka.ru/article/n/sravnitelnyy-obzor-metodov-kategorizatsii-primenyaemyh-v-dlp-sistemah (дата обращения: 06.04.2025).

9. Learn about using regular expressions (regex) in DLP policies [Электронный ресурс]. – URL: https://learn.microsoft.com/en-us/purview/dlp-policy-learn-about-regex-use (дата обращения: 05.04.2025).
10. *Braghin S., Simioni M., Sinn M.* DLPFS: The Data Leakage Prevention FileSystem. DLPFS / arXiv:2108.13785 [cs]. – arXiv, 2021.
11. *Xi N., Chen C., Zhang J., Sun C., Liu S., Feng P., Ma J.* Information flow based defensive chain for data leakage detection and prevention: a survey. Information flow based defensive chain for data leakage detection and prevention / arXiv:2106.04951 [cs]. – arXiv, 2021.
12. *Cao H.* Recent advances in text embedding: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark. Recent advances in text embedding / arXiv:2406.01607 [cs]. – arXiv, 2024.
13. Machine learning: the secret to effective data loss prevention – Polymer DLP [Электронный ресурс]. – URL: https://www.polymerhq.io/blog/machine-learning-the-secret-to-effective-data-loss-prevention/ (дата обращения: 03.04.2025).
14. *Alhindi H., Traore I., Woungang I.* Preventing Data Leak through Semantic Analysis // Internet of Things. – 2021. – Vol. 14. – P. 100073.
15. *Gupta K., Kush A.* A Learning oriented DLP System based on Classification Model / arXiv:2312.13711 [cs]. – arXiv, 2023.
16. Noisy DLP? Regular expressions are probably to blame – Polymer [Электронный ресурс]. – URL: https://www.polymerhq.io/blog/saas-dlp/noisy-dlp-regular-expressions-are-probably-to-blame/ (дата обращения: 04.04.2025).
17. Limitations of data identifier support for PCRE regular expressions – TechDocs [Электронный ресурс]. – URL: https://techdocs.broadcom.com/us/en/symantec-security-software/information-security/data-loss-prevention/16-1/about-data-loss-prevention-policy-authoring/data-identifiers/creating-custom-data-identifiers/limitations-of-data-identifier-support-for-pcre-regular-exp.html (дата обращения: 02.04.2025).
18. *Hassan M., Jincai C., Iftekhar A., Shehzad A., Cui X.* Implementation of Security Systems for Detection and Prevention of Data Loss/Leakage at Organization via Traffic Inspection / arXiv:2012.14111 [cs]. – arXiv, 2020.
19. *Shvartzshnaider Y., Pavlinovic Z., Wies T., Subramanian L., Mittal P., Nissenbaum H.* The VACCINE Framework for Building DLP Systems / arXiv:1711.02742 [cs]. – arXiv, 2017.
20. Is data loss prevention (DLP) relevant in 2025? – Polymer [Электронный ресурс]. – URL: https://www.polymerhq.io/blog/is-data-loss-prevention-dlp-relevant-in-2025/ (дата обращения: 05.04.2025).

СОДЕРЖАНИЕ

Научное издание

СОВРЕМЕННЫЕ МЕТОДЫ, СРЕДСТВА И ТЕХНОЛОГИИ ЗАЩИТЫ ИНФОРМАЦИИ – 2025

Сборник трудов

XVI Международной научно-практической конференции имени Олега Борисовича Макаревича

Таганрог, 19–22 мая 2025 г.

Ответственная за выпуск Е.А. Ищукова

Компьютерная верстка Н.В. Ярошевич

Электронное издание

Подписано к использованию 08.12.2025. Заказ № 10236. Тираж 10 экз.
Усл. печ. л. 6,8. Уч.-изд. л. 5,7.
Издательство Южного федерального университета
Отдел полиграфической, корпоративной и сувенирной продукции
Издательско-полиграфического комплекса КИБИ МЕДИА ЦЕНТРА ЮФУ
344090, г. Ростов-на-Дону, пр-т Стачки, 200/1, тел. (863) 243-41-66