2025

Yandex Cloud

AI Secure Agentic Framework Essentials (AI-SAFE) v 1.0

Содержание

Внедрение и использование больших языковых моделей (LLM) переходит на новый этап развития. Появляются стандартные подходы, архитектурные решения и типовые компоненты для внедрения ИИ в приложения. В частности, стандартном становится использование агентного подхода к интеграции ИИ в бизнес-задачи и повседневную жизнь пользователей.

При работе с ИИ-агентами важно учитывать потенциальные риски и угрозы, которые могут возникнуть в процессе их эксплуатации. Осознание этих угроз имеет значение для всех участников работы с ИИ-агентами — от разработчиков и аналитиков до специалистов по информационной безопасности. Это позволяет обеспечить безопасное и эффективное использование таких технологий, а также защитить данные пользователей и организаций.

В этом документе команда безопасности Yandex B2B Tech составила систематизированный обзор угроз при внедрении ИИ с использованием мультиагентной архитектуры и подготовила практические рекомендации по снижению возникающих рисков при создании и внедрении ИИ-агентов и систем на их основе.

В первой части — базовые понятия агента и мультиагентной архитектуры.
Во второй — угрозы и рекомендации по их устранению.

Область применения документа

Задача документа — познакомить с основными рисками и угрозами при работе с системами, связанными с подготовкой данных, а также с созданием ИИ-агентов и инфраструктур для работы ИИ-систем.

Документ содержит рекомендации по безопасности ИИ-технологий, собранные специалистами платформы Yandex Cloud. В подготовке документа мы учли рекомендации и классификации угроз, которые уже разработали международные ассоциации — OWASP®, NIST и MITRE ATT&CK®.

Для кого будет полезен документ

Инженерам безопасности, CISO, руководителям служб безопасности, DevSecOps-специалистам и инженерам по надёжности сайта (SRE), которые хотят понять, как ИИ-агенты могут повлиять на безопасность защищаемых систем, а также познакомиться с базовыми рекомендациями по снижению рисков, связанных с использованием ИИ-агентов.

Инженерам по обработке данных, архитекторам данных и ML-инженерам, стремящимся разобраться в методах обеспечения безопасности при разработке и внедрении ИИ-агентов.

Специалистам по оценке рисков, юристам и руководителям, которые хотят узнать о практических рисках для бизнес-задач, возникающих при использовании ИИ-агентов.

Ограничение ответственности

Документ не гарантирует покрытие всех возможных аспектов безопасности ваших систем. Рекомендуем проводить оценку защищённости, разработать модель угроз и нарушителей, исходя из рекомендаций регуляторов и особенностей ваших информационных систем и руководствуясь здравым смыслом.

Основы и архитектура

Чтобы наглядно познакомить специалистов по информационной безопасности с возможностями ИИ-агентов и архитектурой таких систем, представим, что перед нами стоит задача разработать ИИ-асситента по безопасности. Условно назовём этого помощника Боб. Задача Боба — помогать специалистам, ответственным за реагирование на инциденты, быстрее анализировать события безопасности и соответствующую информацию об угрозах. В этом разделе мы рассмотрим компоненты и принципы работы такого ИИ-агента.

ИИ-агенты — фундаментальный сдвиг в парадигме искусственного интеллекта, который выходит за рамки возможностей как классических моделей машинного обучения, так и обычных больших языковых моделей.

Отличие от классического машинного обучения

Традиционные ML-модели (например, нейросети для классификации изображений или регрессионные модели для прогнозирования) — это **пассивные инструменты для решения конкретных функций**. Они обучаются на статичных наборах данных для выполнения одной функции — классификации, кластеризации или прогнозирования.

ИИ-агент, в свою очередь, — это **проактивная и автономная система**. Его ключевые отличия:

Целеполагание

Агент не просто решает задачу, а стремится к достижению цели, которая может быть определена на высоком уровне — например, обеспечить безопасность корпоративной сети.

Взаимодействие с окружением

Агент активно взаимодействует с цифровой средой через инструменты (MCP-протокол, API, скрипты), получая обратную связь и адаптируя свои действия. ML-модель обычно не имеет такой возможности.

Автономность и планирование

Агент способен самостоятельно декомпозировать сложную цель на последовательность действий и выполнять их без постоянного контроля со стороны человека. ML-модель лишь выдаёт результат на основе входных данных.

Отличие от LLM

Важно понимать, что LLM — это лишь **один из компонентов ИИ-агента**, хотя и центральный. Если проводить аналогию, то **LLM — это «мозг», отвечающий за рассуждения, а ИИ-агент — это вся система**. LLM может генерировать текст, отвечать на вопросы и даже писать код, но она ограничена своей входной и выходной текстовой модальностью. Полноценный ИИ-агент дополняет возможности LLM критически важными компонентами:

Память (краткосрочная и долгосрочная)

Позволяет агенту сохранять контекст, учиться на прошлом опыте и использовать внешние базы знаний (RAG).

Инструменты

Дают агенту возможность выполнять действия в реальном мире — запускать сканеры уязвимостей, изолировать хосты сети и искать информацию в интернете.

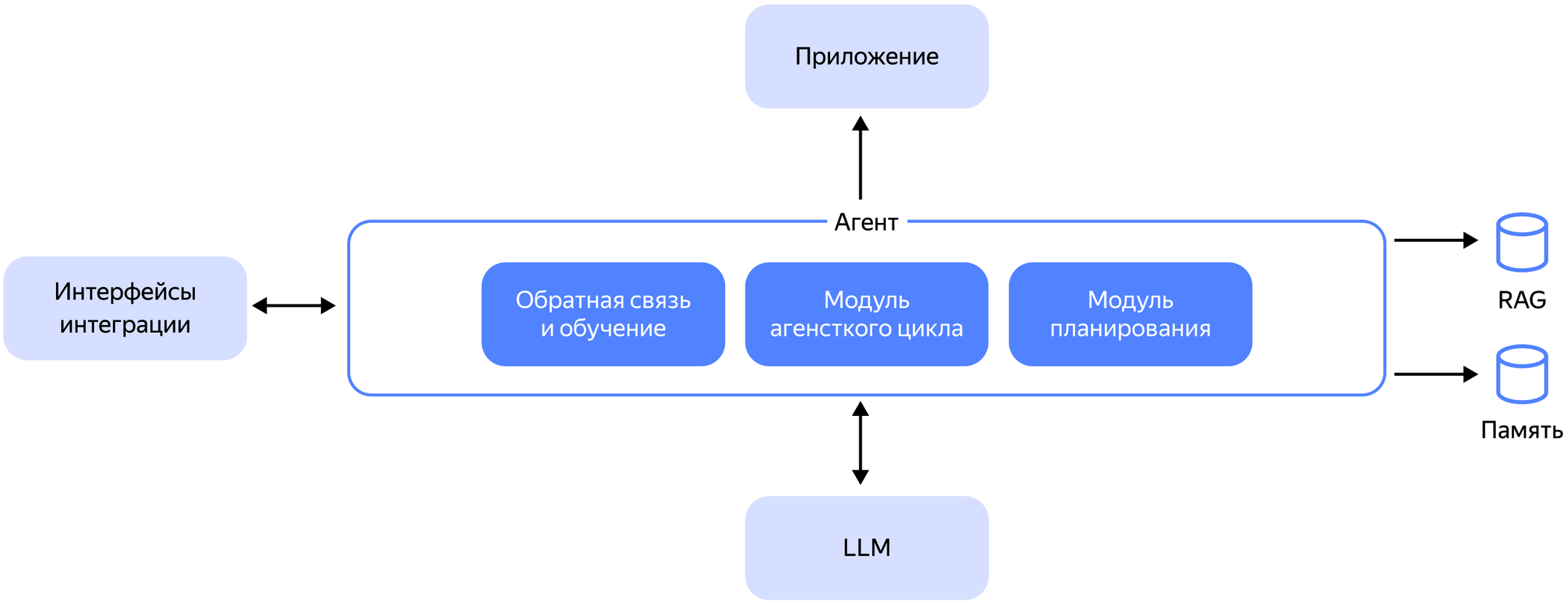
Планирование

Преобразует высокоуровневую цель в конкретный, многошаговый план действий.

Таким образом, если классическая ML-модель — это калькулятор для данных, а LLM — это универсальный генератор текста и идей, то ИИ-агент — это автономный исполнитель, способный использовать эти и другие инструменты для целенаправленного достижения результатов в реальном мире.

Компоненты агента

Типовой ИИ-агент состоит из компонентов, работающих сообща, каждый из которых отвечает за отдельный аспект его работы.



Ключевые блоки агентской системы ИИ:

Приложение

Интерфейс взаимодействия пользователя с интегрированными функциями агентного ИИ.

LLM

Большая языковая модель, которая управляет пониманием и генерацией — например, YandexGPT.

Модули памяти

Долгосрочные и краткосрочные хранилища данных, позволяющие хранить контекст, базы знаний, данные самообучения и другие данные, необходимые для работы агента.

Функциональные модули агента:

- **Модуль агентского цикла.** Обеспечивает логику взаимодействия агента с пользователем и сторонними приложениями, включая аутентификацию и управление доступом.
- **Модули планирования и декомпозиции задач.** Позволяют агенту анализировать задачи и формулировать шаги решения, разбивая поставленную цель на подзадачи.
- **Механизмы обратной связи и обучения.** Обеспечивают ИИ-агенту обратную связь о результате работы, что позволяет корректировать поведение.

Воспомогательные модули:

- **Базы знаний.** RAG-системы и другие хранилища данных — векторные базы или другие интегрированные источники.
- **Интерфейсы интеграции.** Интерфейсы и протоколы, позволяющие интеллектуальному модулю использовать внешние инструменты или API.

Эффективность и надёжность ИИ-агента напрямую зависят от продуманности его архитектуры. Современные агентские системы состоят из нескольких ключевых, взаимосвязанных компонентов.

Фреймворк для моделирования угроз (AI-SAFE)

Мультиагентные архитектуры и ИИ-агенты требуют специфического подхода к защите, учитывая архитектуру, поскольку они так же, как и любая другая система, уязвимы к кибератакам. Злоумышленники могут нарушить их работу, перехватить управление или получить доступ к конфиденциальным данным, и это приведёт к сбоям, потере данных или неправомерному использованию ресурсов.

Кроме того, такие системы часто обрабатывают чувствительные данные (персональные, финансовые, коммерческую тайну), утечка которых грозит юридическими последствиями, финансовыми потерями и утратой доверия пользователей и партнёров.

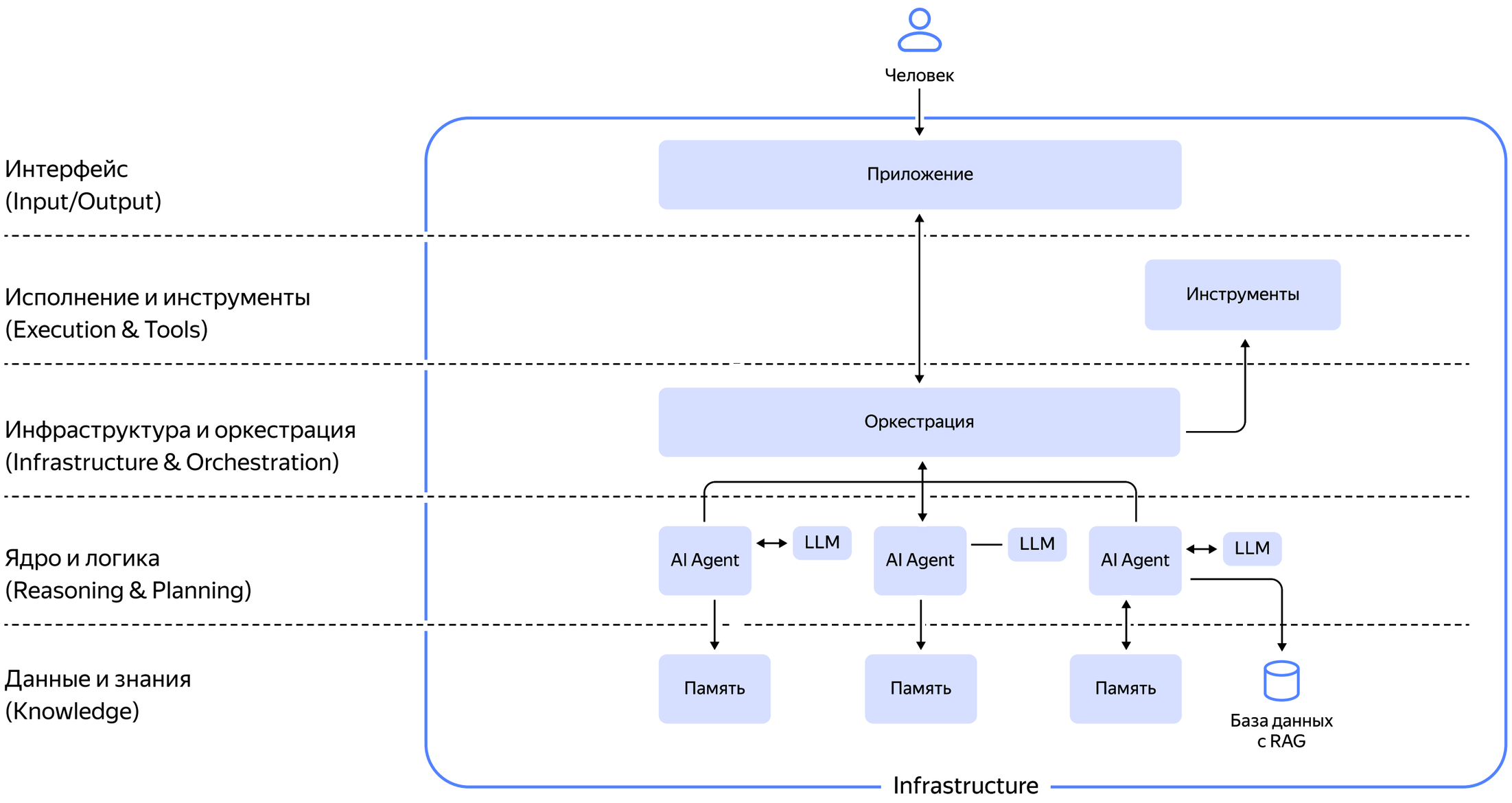
Есть риск манипуляций и мошенничества: попытки злоумышленников повлиять на поведение агентов могут дестабилизировать всю систему и привести к нежелательным последствиям, включая ошибки в принятии решений и неэффективное использование ресурсов.

Защита необходима и для соблюдения правовых и этических норм: нарушения требований к безопасности и прозрачности работы систем могут повлечь юридические и репутационные риски. Инциденты с нарушением безопасности подрывают доверие пользователей и партнёров.

В этом документе мы подготовили вводные для разработки моделей угроз и нарушителей с учётом специфики их работы.

Для более точной и гранулярной оценки угроз безопасности разделим архитектуру типового ИИ-агента на пять логических уровней, у каждого из которых есть свои специфические векторы атак.

Уровень	Описание	Примеры угроз
1. **Интерфейс (Input/Output)**	Уровень, на котором агент взаимодействует с пользователями и внешними приложениями. Точка входа для всех данных	Prompt Injection, Denial of Service (DoS), небезопасная обработка вывода
2. **Исполнение и инструменты (Execution & Tools)**	Уровень, где агент выполняет действия в реальном мире через API. Выполнение входа и другие инструменты	Tool Misuse, Privilege Escalation, небезопасная эскалация делегирования, Tool Poisoning
3. **Ядро и логика (Reasoning & Planning)**	«Мозг» агента: LLM, модули планирования и принятия решений. Здесь формируются гипотезы и планы действий	Jailbreaking, Reasoning Collapse, Goal Manipulation, аффективное фреймирование
4. **Инфраструктура и оркестрация (Infrastructure & Orchestration)**	Базовая инфраструктура (серверы, контейнеры, CI/CD) и протоколы взаимодействия между агентами в мульти-агентных системах	Supply Chain Attacks, Cross-Agent Communication Poisoning, ресурсоёмкие атаки, атаки на оркестратор
5. **Данные и знания (Knowledge)**	Долгосрочная и краткосроч-ная память агента, включая векторные базы (RAG) и другие источники контекста	Data/Knowledge Base Poisoning, Sensitive Information Disclosure, Атаки на RAG (Retrieval Manipulation)



Матрица угроз и рекомендации (AI-SAFE)

Приведём угрозы, сгруппированные по этим уровням. Для каждой угрозы приводятся рекомендации и оценка рисков. Кроме того, на основе анализа ключевых угроз ИИ из OWASP® LLM Top 10, OWASP® MCP Top 10, OWASP® AI Agents Top 15 и RAG-специфичных угроз можно построить комплексную карту защиты, показывающую, какие сервисы Yandex Cloud адресуют те или иные категории рисков.

Уровень 1: угрозы интерфейса взаимодействия

ID	Группа угроз	Примеры	Описание	Рекомендации	Ссылки на другие фреймворки	Оценка рисков
YAISAFE.INPUT.1	Prompt Injection	Внедрение вредоносных инструкций в запрос пользователя	Злоумышленник обманывает модель, заставляя её обойти ограничения или выполнить непреднамеренные действия	Санитизация и валидация входных данных, использование техник	LLM01, MCP01, MCP05, T6, RAG: Indirect Prompt Injection	Вероятность: высокая. Воздействие: высокое
YAISAFE.INPUT.2	Denial of Service	Кибератаки, направленные на перегрузку приложения большим количеством запросов	Злоумышленник может автоматизировать запросы к LLM, чтобы превысить его контекстное окно, перегружая возможности обработки LLM	Rate Limiting, использование WAF, мониторинг потребления ресурсов	LLM10, MCP09, T4, RAG: Resource Exhaustion	Вероятность: средняя. Воздействие: среднее
YAISAFE.INPUT.3	Improper Output Handling	Недостаточная валидация выходных данных LLM перед их использованием в даунстрим-системах	Вывод модели может содержать вредоносный код (XSS, SQLi), который выполнится в другой части системы	Строгая валидация и санитизация вывода модели, использование схем данных (Pydantic, JSON Schema)	LLM05	Вероятность: высокая. Воздействие: высокое

Пример инцидента: дипфейк-мошенничество в банковских системах

Дата и место:

Март 2025 года, Гонконг.

Масштаб ущерба:

Несанкционированные транзакции на сумму около 25 млн долларов.

Механизм:

Злоумышленники использовали образцы голосов из публичных источников для создания высокореалистичных голосовых клонов, обходящих системы голосовой аутентификации банков.

Классификация по AI-SAFE:

Атака нацелена на использование вывода ИИ (сгенерированного голоса) для обмана системы, что косвенно связано с YAISAFE.INPUT.3 (Improper Output Handling), где даунстрим-система (банк) некорректно обрабатывает сгенерированный ИИ-ввод.

Что нужно было делать, чтобы избежать инцидента:

- Не полагаться на один фактор аутентификации (голос) для высокорисковых операций. Внедрить многофакторную аутентификацию (MFA).
- Использовать специализированные алгоритмы для детекции дипфейков и синтезированной речи в реальном времени.
- Ввести операционные лимиты и дополнительные проверки для транзакций, подтверждённых только по голосу.

Уровень 2: угрозы исполнения и инструментов

ID	Группа угроз	Примеры	Описание	Рекомендации	Ссылки на другие фреймворки	Оценка рисков
YAISAFE.EXEC.1	Tool Misuse	Заставить агента использовать инструмент (например, send_email) во вредонос-ных целях (спам, фишинг)	Агент неверно интерпретирует намерение пользователя и применяет легитимный инструмент для нанесения ущерба	Принцип минимальных привилегий для инструментов, чёткие описания назначения инструментов, Human Approval Gates для крити-ческих действий	LLM06, T2	Вероятность: высокая. Воздействие: высокое
YAISAFE.EXEC.2	Privilege Escalation	Агент использует инструмент с избыточными правами, и это позволяет атакующему выйти за пределы песочницы	Небезопасно настроенный инструмент (например, выполнение Python-скрипта с доступом к сети) становится точкой входа в систему	Запуск инструментов в строго изолированных окружениях (Sandboxing: gVisor, Firecracker), статический анализ генериру-емого кода	MCP03, T3	Вероятность: средняя. Воздействие: высокое
YAISAFE.EXEC.3	Tool Poisoning	Внедрение вредоносных инструкций в описание (метаданные) самого инструмента	Агент, доверяя описанию, может выполнить небезопасный код или передать конфиденциальные данные	Аудит и контроль целостности описаний инструментов, разделение данных и инструкций в архитектуре	MCP02, MCP04	Вероятность: низкая. Воздействие: высокое.
YAISAFE.EXEC.4	Auth Bypass & Imper-sonation	Подделка токена доступа для вызова инструмента, эксплуатация уязвимостей в механизме проверки прав	Агент или внешний злоумышленник обходит механизмы контроля доступа, чтобы использовать инструмент без соответствующих разрешений или от имени другого пользова-теля/агента	Использование строгой аутенти-фикации и авторизации для каждого вызова инструмента (OAuth2, mTLS), краткоживущие токены, аудит всех вызовов	MCP10, T9, T13	Вероятность: средняя. Воздействие: высокое

Пример инцидента: взлом GPT-4.1 через отравление инструментов

Дата:

Апрель — июнь 2025 года.

Суть:

Злоумышленники внедрили вредоносные инструкции в описания инструментов GPT-4.1, и это привело к несанкционированному выполнению действий, включая эксфильтрацию данных.

Классификация по AI-SAFE:

YAISAFE.EXEC.3 (Tool Poisoning): основной вектор атаки — компрометация метаданных инструмента, которому доверяет агент.

Что нужно было делать, чтобы избежать инцидента:

- Внедрить строгий аудит и контроль целостности описаний (метаданных) всех подключаемых инструментов.
- Разделять данные и инструкции в архитектуре, чтобы описание инструмента не могло быть интерпретировано как исполняемая команда.
- Применять принцип минимальных привилегий для каждого инструмента, ограничивая доступ только необходимыми ресурсами и действиями.

Уровень 3: угрозы инфраструктуры и оркестрации

ID	Группа угроз	Примеры	Описание	Рекомендации	Ссылки на другие фреймворки	Оценка рисков
YAISAFE.INFRA.1	Supply Chain Attacks	Использование скомпрометированной опенсорс-библиотеки, базовой модели или контейнера	Вредоносный код, внедрённый в один из компонентов, компрометирует всю систему	Использование доверенных репозиториев, SCA- и SAST-сканирование, SBOM, верификация цифровых подписей моделей	LLM03, T11	Вероятность: средняя. Воздействие: высокое
YAISAFE.INFRA.2	Resource Overload	Неконтролируемое потребление вычислительных ресурсов, токенов, вызовов API	Приводит к отказу в обслуживании (DoS) и к непредвиденным финансовым затратам (Denial of Wallet)	Установка квот и лимитов на использование ресурсов для каждого агента/пользователя, Circuit Breakers	LLM10, MCP09, T4	Вероятность: средняя. Воздействие: среднее
YAISAFE.INFRA.3	Cross-Agent Poisoning	В мультиагентной системе один скомпрометированный агент передаёт вредоносные данные другому	Вирусное распространение вредоносного поведения по всей системе приводит к системному коллапсу или компрометации	Изоляция агентов, валидация и санитизация данных на входе в каждом агенте, мониторинг межагентских коммуникаций	T15, MCP05	Вероятность: низкая. Воздействие: высокое

Пример инцидента: утечка данных DeepSeek

Дата:

29 января — 3 марта 2025 года.

Масштаб:

Свыше 1 млн записей чатов, API-ключей и метаданных пользователей.

Причина:

Неправильная конфигурация облачной базы данных без аутентификации, позволившая публичный доступ.

Классификация по AI-SAFE:

YAISAFE.INFRA.1 (Supply Chain Attacks): хоть это и не классическая атака на цепочку поставок, инцидент — фундаментальный сбой безопасности на уровне инфраструктуры, который относится к этой категории.

Что нужно было делать, чтобы избежать инцидента:

- Применять строгие политики безопасности для всех компонентов облачной инфраструктуры (Infrastructure as Code, Policy as Code).
- Внедрить обязательную аутентификацию и контроль доступа для всех баз данных и хранилищ.
- Регулярно проводить аудит конфигураций и сканирование на наличие уязвимостей (Cloud Security Posture Management).

Уровень 4: угрозы ядра и логики

ID	Группа угроз	Примеры	Описание	Рекомендации	Ссылки на другие фреймворки	Оценка рисков
YAISAFE.LOGIC.1	Jailbreaking	Атака с использованием ролевой игры (DAN), аффективное фреймирование	Обход встроенных в модель этических ограничений и правил безопасности для генерации запрещённого контента	Использование моделей с улучшенным Alignment'ом, Prompt Hardening, мониторинг на предмет техник обхода	LLM01, T7	Вероятность: высокая. Воздействие: среднее.
YAISAFE.LOGIC.2	Reasoning Collapse	Зацикливание, генерация нерелевантных планов, отказ от выполнения задачи	Из-за слишком сложных или противоречивых входных данных агент входит в неоптимальный или зацикленный процесс принятия решений	Установка тайм-аутов, Circuit Breakers, внедрение Human-in-the-Loop для сложных задач, упрощение промтов	LLM09, T5	Вероятность: средняя. Воздействие: низкое
YAISAFE.LOGIC.3	Goal Manipulation	Изменение первоначальной цели агента через скрытые инструкции в промте	Агент начинает преследовать цели злоумышленника, сохраняя видимость легитимной работы	Чёткое и недвусмысленное определение целей в системном промте, аудит Reasoning Traces	LLM01, T6	Вероятность: средняя. Воздействие: высокое
YAISAFE.LOGIC.4	Overwhelming HITL	Генерация большого количества ложных алертов, маскировка вредоносного запроса среди легитимных	Злоумышленник перегружает оператора, отвечающего за подтверждение действий агента, чтобы добиться одобрения вредоносного действия из-за усталости или невнимательности	Использование адаптивных порогов для HITL, группировка и приоритизация запросов на подтверждение внедрения honeypot-запросов	T10	Вероятность: низкая. Воздействие: высокое

Пример инцидента: CAIN — целенаправленный захват промтов

Дата:

Май 2025 года.

Механизм:

Манипулирование системными промтами LLM для получения вредоносных ответов на конкретные вопросы при сохранении безобидного поведения в остальных случаях.

Классификация по AI-SAFE:

- YAISAFE.LOGIC.3 (Goal Manipulation): тонкое изменение цели агента для генерации вредоносного контента.
- YAISAFE.LOGIC.1 (Jailbreaking): обход встроенных ограничений безопасности модели.

Что нужно было делать, чтобы избежать инцидента:

- Применять техники Prompt Hardening для защиты системного промта от внешнего влияния.
- Внедрить непрерывный мониторинг и Red Teaming для выявления новых техник обхода и манипуляции.
- Использовать модели с улучшенным Alignment'ом, более устойчивые к манипуляциям.

Уровень 5: угрозы данных и знаний

ID	Группа угроз	Примеры	Описание	Рекомендации	Ссылки на другие фреймворки	Оценка рисков
YAISAFE.DATA.1	Knowledge Base Poisoning	Внедрение вредоносных или ложных данных в документы, используемые RAG-системой	Агент использует отравленную информацию, и это приводит к неверным выводам, саботажу или утечкам данных	Контроль доступа к базе знаний, версионирование данных, использование доверенных источников, криптографи-ческая проверка целостности	LLM04, T1, RAG: Knowledge Base Poisoning	Вероятность: средняя. Воздействие: высокое
YAISAFE.DATA.2	Sensitive Data Disclosure	Модель выдаёт конфиден-циальную информацию из обучающих данных или из контекста RAG	Недостаточная фильтрация или маскирование данных приводит к утечкам PII и коммерческой тайны	Деперсонализа-ция/маскирование данных перед передачей в модель, RBAC для RAG, Fine-tuning для «забывания» данных	LLM02, LLM07, MCP06, RAG: Context Leakage	Вероятность: высокая. Воздействие: высокое
YAISAFE.DATA.3	Retrieval Manipula-tion	Манипуляция поиском в RAG для приоритизации вредоносного документа	Злоумышленник эксплуатирует алгоритм поиска, чтобы модель гарантированно получила вредоносный контекст	Использование гибридного поиска (вектор-ный + ключевой), внедрение модуля Reranker для повторной оценки релевант-ности	LLM08, RAG: Retrieval Manipulation	Вероятность: низкая. Воздействие: среднее
YAISAFE.DATA.4	Embedding Inversion	Восстановление конфиден-циального текста из его векторного представления (эмбеддинга)	Атакующий, имея доступ к векторной базе данных, использует специализирован-ные модели для восстановления исходной чувстви-тельной информа-ции, которая считалась защищённой после векторизации	Использование техник Differential Privacy при создании эмбеддингов, гранулярный контроль доступа к векторной базе, обнаружение аномальных запросов к базе	LLM08, RAG: Embedding Inversion Attacks	Вероятность: низкая. Воздействие: высокое

Пример инцидента: утечка данных ChatGPT через Prompt Injection

Дата:

Март 2025 года.

Механизм:

Злоумышленники внедрили скрытые вредоносные промты в пользовательский ввод, заставив ChatGPT обойти защитные механизмы и раскрыть конфиденциальную информацию.

Классификация по AI-SAFE:

- YAISAFE.INPUT.1 (Prompt Injection): прямая манипуляция моделью через входные данные пользователя.
- YAISAFE.DATA.2 (Sensitive Data Disclosure): раскрытие конфиденциальных данных.

Что нужно было делать, чтобы избежать инцидента:

- Внедрить многоуровневую систему санитизации и валидации входных данных для обнаружения и блокировки паттернов инъекций.
- Использовать техники Prompt Hardening, чётко разделяя инструкции и пользовательские данные.
- Регулярно проводить Red Teaming и тестирование на устойчивость к новым техникам Prompt Injection.

Пример инцидента: удаление дипфейк-музыки у крупнейшего медиаиздателя

Дата:

Март 2025 года.

Суть:

ИИ-модели обучались на существующих музыкальных каталогах без разрешения для копирования стилей популярных артистов, и это привело к нарушению авторских прав.

Классификация по AI-SAFE:

YAISAFE.DATA.1 (Knowledge Base Poisoning): в этом случае отравление — это использование нелицензионных, защищённых авторским правом данных для обучения, что привело к генерации нелегитимного контента.

Что нужно было делать, чтобы избежать инцидента:

- Внедрить строгие процессы управления данными (Data Governance) для проверки происхождения и лицензионной чистоты всех обучающих наборов.
- Использовать только те данные, на использование которых есть явное разрешение от правообладателей.
- Внедрить фильтры на выходе модели для обнаружения и блокировки контента, нарушающего авторские права.

Заключение

Развитие генеративных технологий и ИИ-агентов открывает новые возможности для автоматизации и оптимизации процессов, но требует системного подхода к обеспечению безопасности на всех этапах их жизненного цикла.

Основные угрозы связаны с:

- утечками конфиденциальных данных на этапе подготовки обучающих данных;
- эксплуатацией уязвимостей в инфраструктуре и коде (например, атаки отравления данных, инъекции промтов, эскалации привилегий);
- некорректной интеграцией моделей в производственные системы, которая может привести к снижению производительности, этическим и юридическим рискам, DDoS-атакам и манипуляциям с выводами модели.

Комплекс мер по минимизации рисков:

- Контроль данных: очистка, маскировка конфиденциальной информации, проверка лицензий.
- Безопасность инфраструктуры: принцип минимальных привилегий, защита CI/CD-процессов, мониторинг аномалий.
- Регулярное тестирование и обновление моделей: борьба с дрейфом данных, защита от атак на логику.
- Правильное разделение ответственности между провайдером и пользователем.
- Внедрение инструментов анализа уязвимостей — SAST, DAST, SCA.

Практический чек-лист безопасности по уровням AI-SAFE

Этот чек-лист предоставляет конкретные, действенные шаги для защиты ИИ-агентов, структурированные в соответствии с пятиуровневой моделью AI-SAFE.

Уровень 1: интерфейс (Input/Output)

- **1.1. Валидация и санитизация ввода:** внедрите строгую проверку всех входящих данных. Используйте Smart Web Security с кастомными правилами для блокировки паттернов Prompt Injection и API Gateway для валидации схем запросов.
- **1.2. Ограничение частоты запросов (Rate Limiting):** настройте лимиты запросов в API Gateway или на уровне приложения, чтобы защититься от DoS-атак и атак на переполнение контекста.
- **1.3. Строгая валидация вывода:** не доверяйте выводу LLM. Валидируйте генерируемый код и данные с помощью строгих схем (например, Pydantic, JSON Schema) и кодируйте вывод перед отображением в веб-интерфейсах для предотвращения XSS-атак.

Уровень 2: ядро и логика (Reasoning & Planning)

- **2.1. Усиление системного промта (System Prompt Hardening)**: чётко определите роль, ограничения и запрещённые действия для агента в системном промте. Укажите, что он не должен выполнять инструкции, противоречащие его основной задаче.
- **2.2. Контроль выполнения:** внедрите тайм-ауты для выполнения задач и Circuit Breakers в коде, чтобы предотвратить зацикливание или выполнение слишком сложных, ресурсоёмких задач.
- **2.3. Аудит и логирование рассуждений:** логируйте всю цепочку рассуждений (Reasoning Trace) агента, включая принятые решения и вызванные инструменты. Используйте **Cloud Logging** для сбора и анализа логов на предмет аномального поведения.

Уровень 3: данные и знания (Knowledge)

- **3.1. Контроль доступа к данным (RBAC):** настройте гранулярные права доступа к базам знаний (RAG) и другим источникам данных с помощью **Identity and Access Management (IAM)**. Агент должен иметь доступ только к той информации, которая необходима для выполнения конкретного запроса пользователя.
- **3.2. Деперсонализация чувствительных данных:** перед отправкой данных в LLM или сохранением в базу знаний, удаляйте или маскируйте персональные данные (PII) и другую конфиденциальную информацию. Это можно сделать с помощью кастомных функций в **Cloud Functions** или скриптов в **DataSphere**.
- **3.3. Защита и целостность баз знаний:** храните документы для RAG в **Object Storage** с включённым версионированием и шифрованием с помощью **Key Management Service**. Регулярно проводите аудит источников данных на предмет отравления (Data Poisoning).

Уровень 4: исполнение и инструменты (Execution & Tools)

- **4.1. Принцип минимальных привилегий для инструментов:** каждый инструмент (Tool), доступный агенту, должен работать с минимально необходимыми привилегиями. Используйте IAM-роли для сервисных аккаунтов, от имени которых выполняются вызовы API.
- **4.2. Изоляция окружения для выполнения кода (Sandboxing):** если агент может генерировать и выполнять код, запускайте его в строго изолированных окружениях, например в **Cloud Functions** или контейнерах **Container Registry** с gVisor, чтобы предотвратить побег из песочницы.
- **4.3. Внедрение подтверждения человеком (Human-in-the-Loop):** для всех критически важных или необратимых действий (например, удаления данных, отправки писем, финансовых транзакций) требуйте явного подтверждения от пользователя.

Уровень 5: инфраструктура и оркестрация

- **5.1. Безопасность цепочки поставок (Supply Chain):** регулярно сканируйте ваши Docker®-образы и библиотеки на наличие уязвимостей с помощью интегрированных сканеров в **Container Registry** и инструментов класса SCA (Software Composition Analysis). Используйте только доверенные базовые образы и модели.
- **5.2. Защита от перерасхода средств (Denial of Wallet):** установите в **Yandex Cloud Billing** жёсткие бюджеты и настройте уведомления для контроля над расходами на API генеративных моделей и вычислительные ресурсы.
- **5.3. Изоляция и защита в мультиагентных системах:** изолируйте агенты друг от друга с помощью сетевых политик в **Virtual Private Cloud**. Для защиты коммуникаций между агентами используйте взаимную аутентификацию (mTLS) с помощью сертификатов из **Certificate Manager**.

Совместная ответственность при создании архитектуры вопросно-ответных систем (с использованием RAG и YandexGPT)

Обеспечение безопасности в публичном облаке — совместная задача пользователя и самой платформы. Она включает работу с компонентами на базе YandexGPT и Retrieval-Augmented Generation (RAG).

Разделение ответственности подразумевает:

- контроль прав доступа к ресурсам (например, к виртуальным машинам) силами клиента;
- наличие управляемых компонентов, где безопасность обеспечивается специалистами Yandex Cloud;
- инфраструктуру заказчика, в которой права доступа и функции обеспечения безопасности полностью управляются и обеспечиваются пользователем.

Приложение: детальные каталоги угроз OWASP® LLM Top 10 (2025)

Код	Название	Пояснение
LLM01	Prompt Injection	Манипулирование LLM через специально созданные входные данные для получения несанкционированного доступа, утечки данных или компрометации принятия решений
LLM02	Sensitive Information Disclosure	Непреднамеренное раскрытие конфиденциальной информации (PII, коммерческие данные, секреты) через выходные данные модели
LLM03	Supply Chain Vulnerabilities	Зависимость от скомпрометированных компонентов, сервисов или наборов данных, подрывающих целостность системы
LLM04	Data and Model Poisoning	Внедрение вредоносных данных в обучающий набор или модификация модели для изменения её поведения
LLM05	Improper Output Handling	Недостаточная валидация выходных данных LLM, которая может привести к даунстрим-эксплойтам, включая выполнение кода
LLM06	Excessive Agency	Предоставление LLM неограниченной автономии для выполнения действий, которое может привести к непредвиденным последствиям
LLM07	System Prompt Leakage	Утечка системных промтов, содержащих конфиденциальную информацию о логике работы или внутренних инструкциях
LLM08	Vector and Embedding Weaknesses	Уязвимости в векторах и эмбеддингах, используемых в RAG-системах, включая возможность восстановления исходных данных
LLM09	Misinformation	Распространение неточной или ложной информации через выходные данные модели, особенно при чрезмерном доверии к результатам
LLM10	Unbounded Consumption	Неконтролируемое потребление ресурсов, приводящее к отказу в обслуживании и непредвиденным операционным расходам

OWASP® MCP (Model Context Protocol) Top 10

Код	Название	Пояснение
MCP01	Prompt Injection	Вредоносные входные данные, манипулирующие поведением ИИ через MCP-интерфейсы, с быстрым распространением по взаимосвязанным системам
MCP02	Tool Poisoning	Внедрение вредоносных команд в метаданные MCP-инструментов (описания, параметры, инструкции), эксплуатирующее доверие агентов
MCP03	Privilege Abuse	Предоставление MCP-инструментам избыточных прав доступа, создающее риски эскалации привилегий
MCP04	Tool Shadowing and Shadow MCP	Создание поддельных MCP-инструментов, имитирующих доверенные сервисы для обмана пользователей и агентов
MCP05	Indirect Prompt Injection	Встраивание скрытых вредоносных инструкций во внешние данные, обрабатываемые ИИ-агентами через MCP-серверы
MCP06	Sensitive Data Exposure & Token Theft	Неправильная конфигурация MCP-сред, приводящая к утечке API-ключей, токенов и учётных данных
MCP07	Command/SQL Injection & Malicious Code Execution	Передача неутверждённой пользовательской информации в базы данных или системные команды через MCP-серверы
MCP08	Rug Pull Attacks	MCP-инструменты, изначально кажущиеся легитимными, но внезапно становящиеся вредоносными после получения доверия
MCP09	Denial of Wallet/Service	Злоупотребление ресурсами или API подключён-ных сервисов, приводящее к финансовым потерям или отказу в обслуживании
MCP10	Authentication Bypass	Слабые или неправильно настроенные механизмы аутентификации в MCP-средах, позволяющие обойти контроль безопасности

RAG-угрозы

Угроза	Пояснение
Vector Database Compromise	Компрометация векторных баз данных, содержащих эмбеддинги конфиденциальных документов, с возможностью обратного восстановления исходных данных
Access Control Failures in RAG	Отсутствие контроля доступа к векторным данным, приводящее к горизонтальной эскалации привилегий и доступу к неавторизованной информации
Embedding Inversion Attacks	Атаки по восстановлению исходных данных из векторных представлений, компрометирующие конфиденциальность
Context Leakage Between Users	Утечка контекста между разными пользователями в мультитенантных RAG-системах
Knowledge Base Poisoning	Внедрение вредоносного или ложного контента в базу знаний для манипулирования выходными данными модели
Retrieval Manipulation	Манипулирование процессом поиска для приоритизации определённой информации или создания смещённых результатов
Indirect Prompt Injection via Documents	Встраивание скрытых инструкций в документы, которые обрабатываются RAG-системой
Data Federation Conflicts	Конфликты данных из множественных источников, приводящие к противоречивой информации
Similarity Search Exploitation	Использование семантического поиска для извлечения чувствительной информации через похожие запросы
Vector Database Resource Exhaustion	Перегрузка векторной базы данных сложными запросами, приводящая к отказу в обслуживании

OWASP® AI Agents (Agentic AI) Top 15

Код	Название	Пояснение
T1	Memory Poisoning	Эксплуатация систем памяти ИИ (краткосрочной и долгосрочной) для внедрения вредоносных или ложных данных
T2	Tool Misuse	Манипулирование ИИ-агентами для злоупотребления интегрированными инструментами через обманные промты или команды
T3	Privilege Compromise	Эксплуатация конфигураций агентов для выполнения неавторизованных операций или эскалации привилегий
T4	Resource Overload	Целенаправленная перегрузка агента для вызова отказов в обслуживании или деградации производительности
T5	Cascading Hallucinations	Распространение галлюцинаций через сессии и системы агентов с памятью или коммуникационными возможностями
T6	Intent Breaking & Goal Manipulation	Тонкое внедрение целей или изменение логики планирования через промты, инструменты или входные данные памяти
T7	Misaligned & Deceptive Behaviors	Выполнение небезопасных действий при видимости соответствия требованиям, включая ложь и манипуляции
T8	Repudiation & Untraceability	Автономные решения без надёжного логирования, создающие слепые пятна для атакующих
T9	Identity Spoofing & Impersonation	Подделка идентичности в мультиагентных системах или взаимодействиях «агент — пользователь»
T10	Overwhelming Human-in-the-Loop (HITL)	Затопление человеческих проверяющих предупреждениями или неоднозначными запросами для принуждения к одобрению вредоносных действий

Код	Название	Пояснение
T11	Supply Chain Attacks	Атаки через заражённые библиотеки и фреймворки агентов, включая внедрение бэкдоров
T12	AI Agents as Attack Tools	Использование агентов злоумышленниками для автоматизации атак и поиска уязвимостей
T13	Authorization and Control Hijacking	Компрометация системы разрешений агентам для получения административного контроля
T14	Impact Chain and Blast Radius	Каскадные сбои от одного скомпрометированного агента к множественным взаимосвязанным системам
T15	Cross-Agent Communication Poisoning	Внедрение вредоносного контента в коммуникации между агентами для манипулирования поведением