Как контроль Shadow AI связан с контролем ИИ-агентов

Источник. Блог компании INFERA Security на HABR

https://habr.com/ru/companies/infera_security/articles/1072622/

«У нас ИИ нет, мы только планируем внедрение» – самый обманчивый ответ.
На практике это почти всегда означает: инвентаризации нет, видимости нет, политик нет. При этом сотрудники уже используют публичные модели, IDE-агенты, локальные ассистенты и low-code инструменты с доступом к рабочим данным.

В 2025–2026 годах поверхность атаки сместилась. Если раньше основными рисками были генерация вредоносного контента и утечки через промпт, то сейчас агенты выполняют реальные действия: работают с почтой, CRM, базами данных, API и кодом. Ошибка или компрометация агента – это уже не текстовая проблема, а операционный инцидент.

Классический Shadow AI – это использование сотрудниками ChatGPT, Claude или аналогичных сервисов без ведома ИТ и службы безопасности. **Но большинство LLM Firewall видят только на уровне вызова «теневых» моделей** (API-трафик к моделям) и почти не видят то, что происходит на локальных машинах.

При этом **настоящий Shadow AI живет глубже**: в MCP-инструментах, локальных desktop-агентах, встроенных в корпоративные приложения, в IDE разработчиков и на рабочих станциях сотрудников. Именно поэтому контроль Shadow AI сегодня невозможно рассматривать отдельно от runtime-контроля ИИ-агентов.

От контроля контента к контролю действий

Мы вступили в период, когда технологический ландшафт меняется не по годам, а по неделям. Автономные ИИ-агенты уже не просто генерируют текст, они самостоятельно планируют задачи, выбирают инструменты, обращаются к корпоративным системам, выполняют код и запускают цепочки действий.

В отличие от классических LLM в режиме «вопрос-ответ», агент работает как субъект с правами. Он использует те привилегии, которые ему выдали, и часто имеет доступ сразу к нескольким системам. Если нет сетевой изоляции и контроля поведения, после запуска действие ИИ-агента практически никто не отслеживает: какие операции выполняются, какие данные уходят и с какими узлами устанавливаются соединения.

Это принципиально меняет характер угроз. После первоначальной компрометации (в том числе через indirect prompt injection) агент может продолжать получать инструкции, менять поведение и фактически становиться долгоживущим инструментом злоумышленника. **Ошибка в агентской среде – это уже не ошибка одного продукта, а ошибка на уровне целой категории приложений.**

Многие компании сегодня выбирают LLM Firewall как основное решение для безопасности ИИ. Это понятный и относительно быстрый шаг. Однако если параллельно не внедрять runtime-контроль агентов (перехват вызовов инструментов (tool calls), проверку аргументов и намерений, контроль MCP), организация закрывает только один слой угроз и оставляет открытым самый опасный – **слой реальных действий**.

Пока индустрия обсуждала безопасность моделей и промптов, уязвимая поверхность уже сместилась глубже – в runtime-слой агентских фреймворков и протоколов (включая

MCP). Компании, которые ограничиваются только LLM Firewall, защищают «мозг» агента, но оставляют его «руки» практически без контроля. **А именно руками агент наносит реальный ущерб.**

Почему одного LLM Firewall недостаточно

LLM Firewall решает важную задачу – контролирует взаимодействие с моделями (контроль промптов и ответов, защита данных). Однако ИИ-агент после получения команды переходит к действиям и именно на этом этапе необходимо выстраивать контроль.

Типичные сценарии, которые Firewall почти не закрывает:

- вызов разрешённого инструмента с вредоносными аргументами;
- отравление инструмента и искажение его работы (tool poisoning);
- эксфильтрация данных через легитимный вызов инструмента (tool call);
- отклонение от исходного намерения пользователя;
- использование избыточных привилегий, выданных агенту.

Без перехвата и контроля tool calls, проверки аргументов и сверки с исходным намерением (intent) эти действия просто выполняются.

LLM Firewall контролирует все взаимодействия с ИИ-моделями, а система контроля ИИ-агентов контролирует все действия, которые совершает агент в инфраструктуре компании. **Вместе они закрывают полный цикл: от мысли до действия**. Один слой без другого оставляет критическую дыру. Когда внедрены оба компонента – LLM Firewall и контроль работы ИИ-агентов, то защита становится многослойной и покрывает полный цикл работы агента: от запроса к модели до реального действия в инфраструктуре.

Что должен давать runtime-контроль агентов

Эффективный контроль ИИ-агентов строится на нескольких технических принципах:

1. Обнаружение и инвентаризация (AI-BOM)
Реестр моделей, агентов, MCP-серверов, skills и их связей. Без этого невозможно понять реальный периметр.

2. Runtime-перехват действий
MCP-прокси и agent hooks (PreToolUse / PostToolUse) должны видеть имя инструмента и аргументы **до** выполнения. Вердикт должен выноситься до исполнения.

3. Политики доступа
Всё, что явно не разрешено, должно быть запрещено (Deny-by-default). Ограничения на инструменты, аргументы, пути, домены и объёмы данных.

4. Изоляция и наименьшие привилегии
Агент работает в изолированной среде, со своей идентичностью, без постоянных секретов и с минимально необходимыми правами.

5. Контроль поведения
Журналирование всех вызовов инструментов, пороговые предохранители, эскалация человеку (Human-in-the-Loop (HITL)) для критических действий, аварийный останов и возможность снять forensic-снимок состояния.

6. Fail-closed
При недоступности слоя контроля действие запрещается по умолчанию.

Как это реализовано на практике

(на примере INFERA AI.SafeAgent)

INFERA AI.SafeAgent представляет собой runtime-слой контроля агентов и реализует описанные выше принципы:

- единый список всех обнаруженных агентов с фильтрами по среде, уровню контроля и статусу;
- перехват вызова инструментов (tool calls) с проверкой полномочий и аргументов;
- профили доступа с deny-by-default и least privilege;
- обнаружение Shadow AI с возможностью карантина;
- очередь HITL-одобрений и аварийный останов (Kill Switch);
- полный след сессии для расследования инцидентов;
- режим fail-closed: всё, что явно не разрешено политикой, автоматически запрещено;
- управление ролями (администратор / оператор SOC / аудитор), пользователями и настройками контура.

Система работает совместно с INFERA AI.Firewall как единая платформа, обеспечивая сквозную видимость использования ИИ в корпоративной инфраструктуре компании: от промпта до реального действия ИИ-агента.

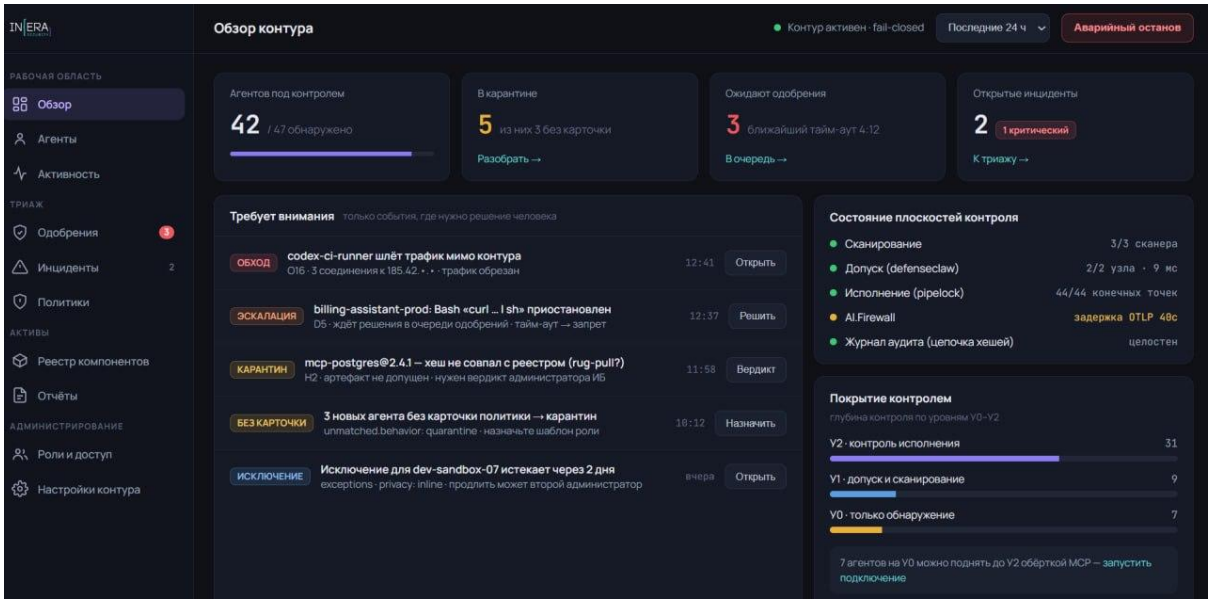

Рис.1. INFERA AI.SafeAgent – аудит, мониторинг и контроль работы ИИ-агентов.

В INFERA AI.SafeAgent в карточке агента представлена вся информация по агенту: метаданные, активность, инциденты, полномочия, доступы, белые списки MCP, криминалистический снимок и т.д.

Единая приборная панель даёт ключевые показатели: агенты под контролем, в карантине, ожидающие одобрения, открытые инциденты, покрытие уровнями контроля, аварийный останов.

Конструктор политик позволяет гибко настраивать правила рисков, полномочия, DLP, сетевые ограничения, MCP-реестр, предохранители, HITL (Human-in-the-Loop) и бюджеты. Система умеет останавливать агента целиком, детектировать зацикливание, работать в fail-closed режиме для необратимых действий и изолировать код в песочнице. Параллельно контролируются затраты через квоты и ML-роутинг.

Решение позволяет закрывать все ключевые угрозы OWASP Top 10 for Agentic Applications 2026 – от Tool Misuse и Privilege Compromise до Rogue Agents. Данный документ прямо подчёркивает принцип Least Agency – не давать агенту больше автономии и полномочий, чем действительно нужно. Это не теоретический список, он построен на реальных инцидентах и синхронизирован с другими материалами OWASP. Рекомендуем CISO и ИБ-командам изучить оригинал этого документа и использовать как живой рабочий инструмент.

Если говорить про контроль Shadow AI, то дополнительно на хосте пользователя реализована многоуровневая проверка: сетевой анализ, контроль процессов, анализ артефактов браузера, обнаружение ключей и токенов, браузерное расширение, которое автоматически различает разрешённые и запрещённые инструменты. Это закрывает самый большой источник перерасхода бюджета и даёт полную видимость, даже если агент запущен локально или через неофициальный канал.

Что важно контролировать уже сейчас

Для CISO и команд ИБ практический минимум выглядит так:

1. вести AI-BOM: модели + агенты + MCP-серверы + skills;
2. разделить контроль моделей и контроль действий агентов;
3. внедрить перехват инструментов вызова (tool calls) до исполнения, а не только логирование после;
4. применять принцип наименьших привилегий к каждому агенту: всё, что явно не разрешено политикой, автоматически запрещено;
5. обеспечить возможность аварийного останова и HITL (Human-in-the-Loop) для критических операций;
6. требовать полный end-to-end след: цепочка рассуждений → вызов инструмента → результат.

Shadow AI сегодня – это не только «сотрудник открыл ChatGPT». Это агенты, которые уже выполняют действия внутри инфраструктуры компании, часто вне поля зрения существующих средств защиты.

Контроль моделей остаётся необходимым, но недостаточным. Без runtime-слоя, который видит и оценивает действия агента, организация защищает только часть угроз и сохраняет ложное чувство безопасности.