



АСТРА ИИ

АСТРА ИИ [ХАБ]

(ASTRA AI Hub)

Руководство администратора

Москва

2026

СОДЕРЖАНИЕ

1. Общие положения.....	3
2. Введение.....	5
3. Установка и первоначальная настройка.....	13
4. Настройка поставщиков и моделей.....	15
5. Безопасность.....	23
6. Администрирование.....	30
7. Типовые проблемы и их решения.....	48
8. Дополнительные возможности.....	62

1. Общие положения

1.1. Область действия руководства

Руководство предназначено для администраторов АСТРА ИИ [ХАБ] (ASTRA AI Hub) (далее — ХАБ). Оно охватывает управление организациями, пользователями, командами и ролями, подписками и ключами, квотами, расписаниями доступа, конфигурацией клиента, моделями и аналитикой.

ХАБ объединяет контур управления доступом и учётом потребления и платформенный контур исполнения моделей. Продуктовый контур определяет доступ к моделям и учёт потребления. Платформенный контур обеспечивает развёртывание и исполнение моделей в Kubernetes, загрузку весов и управление вычислительными ресурсами.

Консоль выполняет операции через продуктовый API. Серверная часть передаёт изменения ресурсов моделей через HTTP-адаптер, а оператор Kubernetes приводит фактическое состояние к заданному.

Внешний клиент — приложение, которое обращается к API ХАБ. Оно может находиться внутри сети заказчика. Установка, обновление, лицензирование и локальное хранение данных клиентов АСТРА ИИ [КОД] (далее – AstraCode) — АстраCLI, расширения редактора и настольного приложения — не входят в область этого руководства. Агентский цикл и выполнение инструментов относятся к клиентскому приложению.

1.2. Термины и определения

Термин	Определение
Организация	Единица изоляции продуктового контура, с которой связаны пользователи, роли, подписка и лимиты.
Команда	Группа пользователей внутри организации; область применения политик.
Пользователь	Участник продуктового контура, идентифицируемый системой идентификации. Допускается не более одной активной привязки к организации.
Инфраструктурный арендатор (Tenant)	Представление продуктовой организации в платформенном контуре.
Клиент API	Инфраструктурная сущность доступа к моделям, связанная с персональным ключом пользователя либо сервисным ключом организации.
Конфигурация клиента	Централизованные настройки поставщика и моделей для клиентского приложения.
Конфигурация развёртывания	Описание параметров размещения и исполнения модели.
Развёртывание модели	Ресурс модели, который регистрируется в Kubernetes через HTTP-адаптер; его фактическое состояние согласует оператор.

Термин	Определение
	Также используется термин «деплой».
Квота	Допустимый объём потребления при обращении к модели. Показатели и область действия задаются тарифом и настройками.
Подписка	Предоставление организации доступа к моделям на условиях тарифа. Активная подписка необходима для запросов и каталога моделей организации.
Тариф	Условия использования: разрешённые модели, ограничения пользователей и сервисных ключей, квоты и цена периода.
Ключ доступа к моделям	Секрет для запросов к моделям. Персональный ключ связан с пользователем; сервисный предоставляет машинный доступ организации без привязки к пользователю и команде.

Выбор модели в конфигурации клиента не создаёт развёртывание и не выделяет GPU. Сохранение конфигурации развёртывания и регистрация ресурса не означают готовность модели обслуживать запросы.

Подписка регулирует потребление моделей, а не лицензирование программного обеспечения ХАБ.

1.3. Сокращения и обозначения

Обозначение	Значение
ИИ	Искусственный интеллект.
API	Программный интерфейс взаимодействия приложений.
OIDC	OpenID Connect — протокол аутентификации через систему идентификации.
GPU	Графический процессор для вычислений, в том числе исполнения моделей.
SSE	Server-Sent Events — передача событий от сервера клиенту; используется для потоковой выдачи ответа.
Консоль ХАБ	Интерфейс администрирования.
Серверная часть ХАБ	Продуктовый API и политики
HTTP-адаптер	Компонент взаимодействия серверной части с ресурсами Kubernetes.
Оператор моделей	Компонент Kubernetes, согласующий фактическое состояние ресурсов моделей с заданным.
Шлюз запросов	Маршрутизация запросов к локальным и

Обозначение	Значение
	внешним моделям через OpenAI-совместимый API.

OpenAI-совместимость означает совместимость интерфейса в пределах поддерживаемых операций, а не обязательное обращение к сервису OpenAI.

1.4. Эксплуатационные границы

Базовое размещение ХАБ — Kubernetes-кластер заказчика. Локальные модели используют его ресурсы; внешние модели обрабатывают данные у подключённого поставщика. Размещение ХАБ у заказчика не исключает внешнюю передачу данных.

Клиентские обращения проходят продуктовые проверки доступа, подписки и квот. Прямое подключение к внутреннему шлюзу не используется для обхода этих проверок. Служебные интерфейсы не публикуются как пользовательские точки входа.

Общий шлюз, PostgreSQL и ValKey обслуживают разные функции контуров.

Дополнительные подсистемы используются в пределах включённой конфигурации. Отсутствие необязательной подсистемы мониторинга не означает отказ модели и не отменяет обязательные зависимости продуктового учёта.

2. Введение

2.1. Назначение АСТРА ИИ [ХАБ]

АСТРА ИИ [ХАБ] — корпоративная платформа управления доступом к моделям искусственного интеллекта и их эксплуатацией. ХАБ объединяет доступ к локальным и внешним моделям, правила их использования, управление развёртываниями и учёт потребления. Пользователи и приложения обращаются к разрешённым моделям через единый программный интерфейс, совместимый с OpenAI.

Для администратора в ХАБ предусмотрены две взаимосвязанные области работы:

- **Управление доступом и потреблением:** организации, пользователи, команды, роли, подписки, ключи запросов, квоты, расписания доступа, централизованная конфигурация клиента и аналитика.
- **Управление эксплуатацией моделей:** конфигурации и развёртывания, состояние моделей и сведения о вычислительных ресурсах.

Организация служит единицей изоляции пользователей, прав, подписки и лимитов. Команда объединяет пользователей внутри организации и может быть областью применения политик. Инфраструктурные сущности, соответствующие организации и клиентам программного интерфейса, обслуживаются платформенным контуром; их не следует отождествлять с пользовательскими ролями.

Базовый вариант размещения — Kubernetes-кластер заказчика. Для локальных моделей используются вычислительные ресурсы этого контура; внешние модели вызываются через подключённого поставщика. Наличие единой точки доступа само по себе не означает, что обработка всех запросов выполняется внутри инфраструктуры заказчика.

Первоначальный экран консоли

На первоначальном экране «Админ-панель» открыт раздел «Организации». В левой части расположено меню: «Организации», а в группе «ПЛАТФОРМА» — «Квоты», «Подписки», «Сервисные API-ключи» и «Аналитика».

Организации представлены карточками. Карточка содержит название и идентификатор организации, её статус, количество пользователей и дату создания. В нижней части боковой панели отображается текущая учётная запись и кнопка «Выйти».

Откройте карточку нужной организации. Внутри доступны вкладки «Пользователи», «Роли», «Команды», «Расписания доступа» и «Конфигурация клиента». Для возврата к списку используйте ссылку «Организации» над названием. Перед изменениями сверяйте выбранную организацию; название группы меню «ПЛАТФОРМА» само по себе не подтверждает полномочия администратора платформы.

Раздел	Для чего использовать
«Пользователи», «Роли», «Команды»	Управлять составом организации, назначениями и группировкой пользователей. Порядок работы приведён в разделах 6.1 и 6.2.
«Конфигурация клиента»	Задать поставщика, основную модель, модель подчинённого агента и модели ролей клиента. Порядок настройки приведён в разделе 4.1.
«Расписания доступа»	Задать временные окна доступа к запросам. Это не расписание остановки экземпляров модели.
«Подписки» и «Квоты»	Проверить действующий тариф и ограничения использования выбранной модели.
«Сервисные API-ключи»	Создать или отозвать ключ машинного доступа организации.
«Аналитика»	Проверить потребление и ошибки, выбрать период и фильтры, выгрузить CSV.

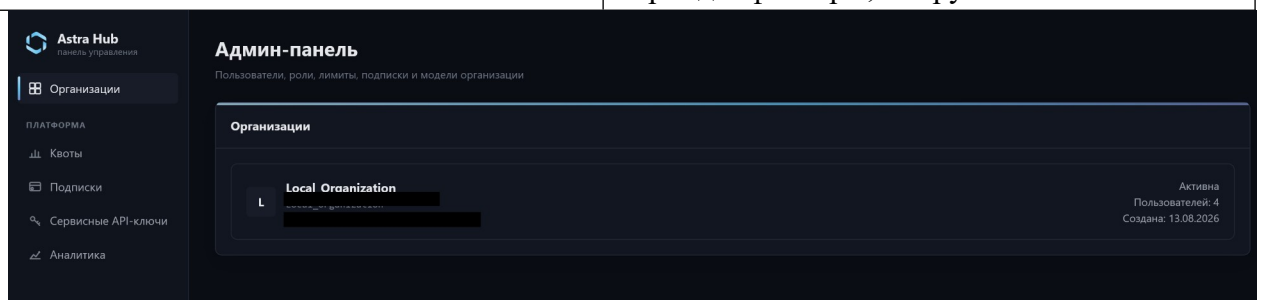


Рисунок 1 — Главный экран административной панели

ХАБ и клиентские приложения

ХАБ централизованно хранит конфигурацию клиента: для неё предусмотрены области «Организация», «Команды» и «Пользователи». При отсутствии отдельной конфигурации интерфейс сообщает о применении конфигурации вышестоящего уровня. Выбор модели

здесь настраивает её использование клиентом, а не создаёт развёртывание и не выделяет GPU.

Агентский цикл, выполнение инструментов и управление подчинёнными агентами относятся к клиентскому приложению. В AstraCode их выполняет общее ядро клиентов АстраCLI, Plugin и Desktop. Поля Explorer и Worker в ХАБ задают модели ролей клиента; они не предоставляют административных прав. Настройки клиента и его собственные правила исполнения не заменяют серверные проверки подписки, расписания и квот ХАБ.

2.2. Состав системы и принцип работы

ХАБ включает продуктовый контур управления, платформенный контур исполнения моделей и соединяющие их компоненты. Для эксплуатации важно различать операцию управления — например, изменение конфигурации развёртывания — и запрос, при котором модель обрабатывает данные клиента.

Компонент	Назначение
Консоль управления	Рабочий интерфейс администратора. Обращается к продуктовому программному интерфейсу, а не непосредственно к Kubernetes.
APISIX	Входной шлюз: проверяет подлинность пользователя или сервисного ключа и направляет обращения к компонентам ХАБ. Заменяет недоверенные клиентские заголовки идентификации доверенными.
Система идентификации Dex(Keycloak)	Обеспечивает идентификацию пользователей и интерактивный вход по протоколу OIDC.
Серверная часть	Обрабатывает операции управления, проверяет права и продуктовые политики, ведёт организации, подписки, ключи, конфигурации моделей и учёт.
HTTP-адаптер	Передаёт операции серверной части в Kubernetes, преобразуя обращения программного интерфейса в операции с пользовательскими ресурсами.
Шлюз запросов к моделям	Маршрутизирует разрешённые запросы к локальным средам исполнения или внешним поставщикам.
Kubernetes-оператор моделей	Приводит фактическое состояние ресурсов к заданному: управляет средами исполнения моделей, томами и состоянием развёртываний.
Загрузчик и кэш весов	Обеспечивают загрузку и хранение файлов параметров локальных моделей.
ValKey и PostgreSQL	Поддерживают определение ключей, резервирование квот, передачу событий

Компонент	Назначение
	учёта и хранение продуктового журнала. Их назначения различаются: ValKey обслуживает оперативные данные, PostgreSQL — долговременный журнал.
Дополнительные платформенные компоненты	Обеспечивают масштабирование, сбор метрик, хранение диалогов и защиту содержимого в пределах включённых функций.

Путь операции управления

Операции администратора выполняются по следующей схеме:

1. Администратор обращается к консоли и проходит проверку подлинности через входной шлюз и систему идентификации.
2. Консоль направляет обращение к продуктовому программному интерфейсу. Серверная часть проверяет полномочия на запрошенную операцию.
3. Если операция затрагивает ресурсы моделей в Kubernetes, серверная часть обращается к HTTP-адаптеру. Операции только с продуктовыми данными не требуют обращения к кластеру.
4. Для развёртывания модели адаптер регистрирует соответствующий ресурс, а оператор приводит фактическое состояние к заданному.
5. Сведения о состоянии развёртывания поступают администратору через продуктовый программный интерфейс.

Прямые операции консоли с Kubernetes не предусматриваются. Регистрация ресурса модели ещё не означает готовность модели принимать запросы: после неё выполняются платформенные операции, в том числе подготовка весов и среды исполнения. При оценке результата изменения необходимо различать принятие операции и достигнутое состояние развёртывания.

Путь запроса к модели

Продуктовый путь запроса строится по схеме «проверка доступа → резерв квоты → вызов → фиксация потребления»:

1. **Проверка подлинности.** APISIX проверяет пользовательское обращение либо сервисный ключ. Сведения о вызывающем формируются на доверенной стороне системы.
2. **Проверка продуктовых условий.** До вызова модели проверяются применимые условия: членство пользователя в организации, активная подписка, разрешение на модель, расписание доступа и квота. Для сервисного ключа действуют политики организации, без пользовательского и командного уровня.
3. **Резервирование квоты.** Перед передачей запроса резервируется допустимый объём потребления. При отказе в резервировании модель не вызывается.
4. **Выполнение.** Разрешённый запрос передаётся через продуктовую точку входа `/api/v1/inference/{path}` в шлюз моделей, затем — локальной или внешней модели.

5. **Ответ и учёт.** Клиент получает результат, а фактическое потребление фиксируется в продуктовом учёте. Для передачи событий используется ValKey, для записи продуктового журнала — PostgreSQL с фоновым обработчиком.

Отказ по правилам доступа или квоте следует отличать от неисправности модели: такой отказ происходит до её вызова. Отсутствие активной подписки закрывает запросы к моделям и каталог моделей организации, но не снимает права управления организацией.

Эта схема относится к продуктовому пути с проверкой политик и учётом. Использовать прямое подключение к нижестоящему шлюзу как обход продуктовых проверок нельзя; иные схемы доступа допустимы только в пределах согласованной конфигурации поставки.

Обязательные и дополнительные компоненты

- **Платформенная основа:** Kubernetes, средства установки Helm, оператор, шлюз запросов и пользовательские ресурсы жизненного цикла моделей. Этот состав может работать без дополнительных подсистем платформенного учёта, диалогов, защиты содержимого, KEDA и VictoriaMetrics.
- **Продуктовый режим:** для консоли и пользовательского входа требуется система идентификации; для продуктового учёта и аналитики необходим PostgreSQL. ValKey обязателен в промышленном продуктовом режиме. Необязательность платформенного учёта не означает, что можно исключить продуктовый журнал или резервирование квот.
- **Компоненты по функции:** S3-совместимое хранилище применяется для соответствующих задач хранения; KEDA и источник метрик — для масштабирования числа реплик; средства мониторинга — для включённого наблюдения за инфраструктурой.
- **Дополнительные средства защиты и доступа:** проверка содержимого Guardrails и Termidesk Connect включаются отдельно. Termidesk Connect не относится к обязательному минимуму. Защита содержимого по умолчанию выключена; её наличие в составе продукта не означает, что запросы уже проверяются или маскируются.

Перед эксплуатацией следует установить, какие дополнительные функции включены. Отсутствие необязательного компонента нельзя автоматически считать неисправностью; при этом отключённая функция не должна учитываться как действующая мера контроля или защиты.

2.3. Администраторы и области полномочий

Права управления проверяет серверная часть ХАБ. Успешный вход через APISIX подтверждает подлинность обращения, но сам по себе не предоставляет административных полномочий.

В продуктовой модели разделены администратор платформы и администратор организации. Это области полномочий пользователей ХАБ, а не названия команд разработки и не права доступа к Kubernetes.

Администратор платформы

К области администратора платформы относятся:

- ведение конфигураций развёртывания моделей;

- управление развёртываниями и связанными платформенными операциями через продуктовый интерфейс;
- ведение каталога тарифов;
- доступ к аналитике любой организации.

Членство в конкретной организации для платформенных полномочий не обязательно. Назначение роли внутри организации не заменяет предоставление прав администратора платформы.

Администратор организации

Встроенная защищённая роль администратора организации предусматривает полный доступ в пределах своей организации. К этой области относятся управление пользователями, командами, ролями, ключами и квотами организации, а также просмотр её аналитики.

В консоли встроенная роль называется Organization Admin и отмечена как «системная роль». Разрешения остальных ролей проверяйте по их составу на вкладке «Роли», а не по названию. Например, название Viewer само по себе не означает доступ ко всем разделам на чтение.

Роли организации формируются из фиксированного каталога разрешений. Они не дают права управления конфигурациями и развёртываниями моделей на уровне платформы. Управление квотами организации также не означает право менять общий каталог тарифов или игнорировать ограничения подписки.

При работе необходимо различать три условия: назначенные полномочия, состояние подписки и доступность функции в поставке. Наличие административной роли не отменяет ограничений доступа к моделям и не добавляет отсутствующие операции.

2.4. Задачи и подготовка администратора

Регулярные работы распределяются по области полномочий. Их периодичность и ответственные лица закрепляются эксплуатационным регламентом организации.

Область работ	Администратор организации	Администратор платформы
Пользователи и права	Поддерживает актуальность состава пользователей, команд и разрешений своей организации.	Разбирает обращения, требующие платформенных полномочий; не подменяет организационные роли правами платформы.
Ключи и доступ к моделям	Контролирует ключи своей организации, состояние подписки и применимые ограничения.	Поддерживает каталог тарифов и согласованный состав доступных моделей в пределах реализованных операций.
Потребление	Анализирует использование и отказы в своей организации; различает исчерпание квоты и техническую ошибку.	Анализирует использование организаций и обращения по продуктовым ограничениям.

Область работ	Администратор организации	Администратор платформы
Эксплуатация моделей	Передаёт сведения о недоступности модели администратору платформы.	Сопровождает конфигурации и развёртывания, оценивает их состояние; привлекает ответственных за инфраструктуру при платформенных неисправностях.
Разбор отклонений	Указывает затронутую организацию, модель, время и наблюдаемый результат, не передавая секретные ключи.	Сопоставляет продуктовые сведения с состоянием платформы и направляет обращение ответственным за нужный контур.

Для выполнения этих работ администратору организации следует понимать модель организаций, команд и ролей, различия персонального и сервисного доступа, действие подписки и квот, а также правила безопасного обращения с ключами и данными запросов.

Администратору платформы дополнительно необходимы знания жизненного цикла моделей в Kubernetes, назначения HTTP-адаптера и оператора, зависимости готовности модели от весов, среды исполнения и вычислительных ресурсов. Понимание назначения PostgreSQL и ValKey требуется для разбора нарушений продуктового учёта; понимание метрик и журналов — для взаимодействия с инфраструктурной эксплуатацией.

Перед началом работ администратору необходимо получить:

- сведения об установленном выпуске и включённых функциях;
- назначенные полномочия и перечень организаций, за которые он отвечает;
- утверждённый способ входа в консоль;
- перечень доступных моделей, действующих тарифов и ограничений;
- правила обработки служебных данных и секретов;
- порядок передачи обращений по продуктовому, платформенному и интеграционному контурам.

Перечисленные знания и подготовительные действия служат ориентиром для организации эксплуатации. Формальные требования к квалификации, порядок назначения администраторов и обязанности по сопровождению инфраструктуры устанавливаются заказчиком совместно с ответственными за поставку. Продуктовая роль администратора платформы сама по себе не предоставляет доступ к кластеру, базам данных или системе идентификации.

2.5. Взаимодействие контуров управления и исполнения

Для эксплуатации следует различать продуктовый контур управления и платформенный контур исполнения моделей внутри единого ХАБ. Первый определяет правила доступа и учёт, второй обеспечивает техническое выполнение операций с моделями.

Контур	Область ответственности
Астра ИИ — продуктовый контур	Консоль, идентификация, организации, пользователи и роли; тарифы, подписки и ключи; проверки доступа до вызова модели; продуктовые квоты и аналитика.
Платформенный контур	Kubernetes-оператор, пользовательские ресурсы, загрузка и кэш весов, среды исполнения моделей, управление GPU и масштабированием, инфраструктурные метрики и журналы.
Совместный слой	Шлюз запросов к моделям, интеграционные контракты и совместно используемые хранилища, и средства передачи событий.

В совместном слое ответственность разделяется по назначению, а не только по имени компонента:

- **Шлюз запросов.** Продуктовые функции учёта, ограничений и расписаний относятся к контуру Астра ИИ; инфраструктурная поставка шлюза — к платформенному контуру.
- **PostgreSQL.** Продуктовый журнал и дополнительные платформенные данные учёта и диалогов имеют разные назначения, схемы и роли доступа. Наличие общей технологии хранения не объединяет полномочия их администраторов.
- **ValKey.** Хранит оперативные данные ключей, резервы квот и события продуктового учёта.
- **HTTP-адаптер.** Это граница между продуктовым управлением и ресурсами Kubernetes. Администратор работает через ХАБ, а не обращается к адаптеру как к самостоятельной консоли.

Персональный владелец сопровождения HTTP-адаптера и защиты содержимого должен быть закреплён в эксплуатационном регламенте поставки: их ответственность затрагивает оба контура. Разделение команд разработки не заменяет назначения ответственных за эксплуатацию у заказчика.

При недоступности модели сначала следует определить характер проблемы. Отказ из-за прав, подписки или квоты относится к продуктовым условиям. Ошибка загрузки весов, нехватка GPU или неготовность среды исполнения — к платформенной эксплуатации. Если продуктовая операция принята, но нужное состояние не достигнуто, разбор может потребовать участия обоих контуров и проверки обмена через HTTP-адаптер. Для передачи обращения используются сведения об организации, модели, времени, результате операции и идентификаторе запроса, если он доступен; секретные ключи в обращение не включаются.

3. Установка и первоначальная настройка

3.1. Размещение и зависимости

Базовое размещение ХАБ — Kubernetes-кластер заказчика. Платформенная основа использует Kubernetes версии не ниже 1.29 и Helm. Нижняя граница версии не означает совместимость с любой более новой версией. Для установки используется совместимый комплект компонентов и зависимостей одного выпуска.

Зависимость	Назначение
Kubernetes и Helm	Размещение компонентов и установка платформы.
PostgreSQL	Продуктовый журнал, учёт и аналитика.
ValKey	Кэш ключей, контекст запросов, резервы квот и события учёта. Обязателен в промышленном продуктивном режиме.
Система идентификации	Вход в консоль и аутентификация OIDC-клиентов.
S3-совместимое хранилище	Хранение весов и данных соответствующих включённых подсистем.
KEDA и Prometheus либо VictoriaMetrics	Масштабирование числа экземпляров модели при включённой функции.

Вычислительные ресурсы локальных моделей учитываются отдельно от ресурсов компонентов управления и хранения. При обновлении модели новая реплика достигает готовности до удаления старой; для такой замены требуется запас ресурсов.

Отсутствие платформенного хранения диалогов не отменяет PostgreSQL для продуктовой аналитики. Все данные ValKey нельзя считать удаляемым кэшем: в нём находятся также резервы и события учёта.

3.2. Комплект и конфигурация установки

Установка выполняется средствами Helm по регламенту установленного выпуска. В состав платформы входят ресурсы Kubernetes, оператор, шлюз, загрузчик, интерфейс и включённые опции. Версии компонентов, конфигурация и зависимости фиксируются в эксплуатационной документации.

Установочные операции выполняются только в целевом окружении с разрешёнными правами. Пакеты, образы и веса получают из разрешённых источников. Установку без регламента и совместимого комплекта не выполняют.

Секреты источников весов и внешних моделей передаются через Kubernetes Secret. В спецификациях ресурсов используются ссылки на секреты, а не открытые значения. Значения секретов не включаются в журналы работ и диагностические материалы.

Роли и схемы PostgreSQL продуктового и платформенного контуров разграничиваются. Миграции выполняются средствами соответствующего выпуска. Ручное создание предполагаемых таблиц и запуск миграций другой версии запрещены.

Постоянные тома и ресурсы моделей не удаляют ради повторного запуска установки. При ошибке фиксируют этап, фактическое состояние и диагностические сведения без секретов; дальнейшие действия выполняют по регламенту восстановления.

3.3. Режимы серверной части

Режим	Назначение	Зависимости готовности
Только управление	Консоль, организации, развёртывания и политики	HTTP-адаптер; база данных и ValKey для учёта и кэша.
Только исполнение	Приём и выполнение	Шлюз моделей; база

Режим	Назначение	Зависимости готовности
запросов	запросов к моделям	данных и ValKey для учёта и кэша.
Совмещённый	Управление и исполнение	Зависимости обоих режимов.

Успешный запуск процесса не означает готовность его зависимостей. Отключение обязательной зависимости или проверки готовности не является способом завершить установку.

3.4. Первоначальный доступ

Интерактивный вход проходит через APISIX и систему идентификации по OIDC с потоком Authorization Code и PKCE. APISIX устанавливает доверенную личность, а серверная часть проверяет полномочия на операцию.

Клиентские заголовки X-User-* заменяются доверенными значениями. Серверная часть и HTTP-адаптер не публикуются как обходные пользовательские интерфейсы. Продуктовая точка запросов к моделям — `/api/v1/inference/{path}`.

Права администратора платформы отделены от ролей организации. Назначение роли организации не предоставляет платформенных полномочий. Прямая правка базы данных не используется для назначения первого администратора.

На первоначальном экране консоли открыт раздел «Организации». Выберите карточку организации и сверьте её название и идентификатор. В карточке доступны «Пользователи», «Роли», «Команды», «Расписания доступа» и «Конфигурация клиента». Порядок работы с составом и правами приведён в главе 6.

Для пользовательского запроса требуются членство в организации, активная подписка, разрешённая модель, подходящее расписание доступа и доступная квота. Для сервисного ключа применяются политики организации без пользовательского и командного уровня. Выбор моделей клиентского приложения описан в разделе 4.1, проверка доступа — в разделе 4.6.

3.5. Контроль первоначального запуска

Проверка `/health/live` показывает, что процесс запущен, и не зависит от внешних сервисов. Проверка `/health/ready` оценивает зависимости выбранного режима. Служебные проверки выполняют из разрешённой точки; наружу их не публикуют.

Доступность страницы входа не заменяет проверку входа администратора, разграничения прав и запросов к модели. Принятие операции развёртывания не означает готовность модели: оператор подготавливает веса, тома и экземпляры исполнения.

После первоначальной настройки выполните проверку доступа по разделу 4.6 и сопоставьте запрос с аналитикой по разделу 6.6. Успех запроса к внешней модели не подтверждает работу локальных GPU. При отсутствии стоимости или категорий токенов сохраняйте отсутствие сведений, не подставляя нулевые значения.

В протоколе запуска фиксируются выпуск, конфигурация, исполнитель, время, выполненные действия и фактические результаты. Производственный доступ открывается после успешного контроля обязательных зависимостей, входа, границ доступа и продуктового учёта. Ключи и содержимое пользовательских запросов в протокол не включаются.

4. Настройка поставщиков и моделей

Настройка моделей включает две разные задачи: выбор поставщика и моделей для клиентского приложения и управление исполнением моделей. Первую выполняют в организации на вкладке «Конфигурация клиента». Конфигурации развёртывания, локальные развёртывания и внешние подключения относятся к полномочиям администратора платформы. Роли организации не предоставляют эти платформенные права.

При развёртывании серверная часть ХАБ передаёт операции HTTP-адаптеру, а оператор Kubernetes подготавливает веса, тома и экземпляры исполнения. Консоль работает через продуктовый программный интерфейс (API), а не непосредственно с Kubernetes.

4.1. Конфигурация клиента и конфигурации развёртывания

Различайте следующие объекты.

Объект	Назначение	Проверка результата
Модель в каталоге организации	Модель, доступная для выбора в контексте подписки организации	Убедиться, что она разрешена организации и отвечает на запрос
Конфигурация клиента	Поставщик и модели, предназначенные для использования клиентским приложением	Проверить сохранённые значения и фактические настройки клиента
Конфигурация развёртывания	Спецификация размещения и исполнения модели	Проверить параметры и предварительный просмотр без создания ресурса
Развёртывание	Ресурс модели, фактическое состояние которого согласует оператор	Проверить состояние, затем контрольный запрос

Выбор модели в конфигурации клиента не создаёт развёртывание и не выделяет GPU. Список выбора также не заменяет каталог конфигураций развёртывания или историю использования в аналитике.

Подготовка к настройке клиента

1. Определите организацию и требуемую область настройки: организация, команда или пользователь.
2. Проверьте действующую подписку в разделе «Подписки». Без активной подписки недоступны запросы и каталог моделей организации, в том числе для конфигурации клиента; права управления организацией сохраняются.
3. Подготовьте согласованные идентификатор и отображаемое имя поставщика, базовый адрес подключения клиента, основную модель и модель подчинённого агента. При необходимости подготовьте модели Explorer и Worker и минимальную версию командного клиента.
4. Для адреса подключения используйте утверждённую точку входа поставки. Подсказка адреса в форме — пример, а не рабочий адрес ХАБ. Не подставляйте адрес Kubernetes, HTTP-адаптера или внутреннего шлюза вместо продуктовой точки входа.

Создание конфигурации организации

1. Откройте «Организации» и карточку нужной организации.
2. Перейдите на вкладку «Конфигурация клиента», затем выберите вложенную вкладку «Организация».
3. Проверьте текущее состояние. Если отдельная конфигурация отсутствует, показано сообщение: «Для этой области отдельная конфигурация не задана. Будет применена конфигурация вышестоящего уровня.» Чтобы задать собственные значения для организации, нажмите «Создать конфигурацию».

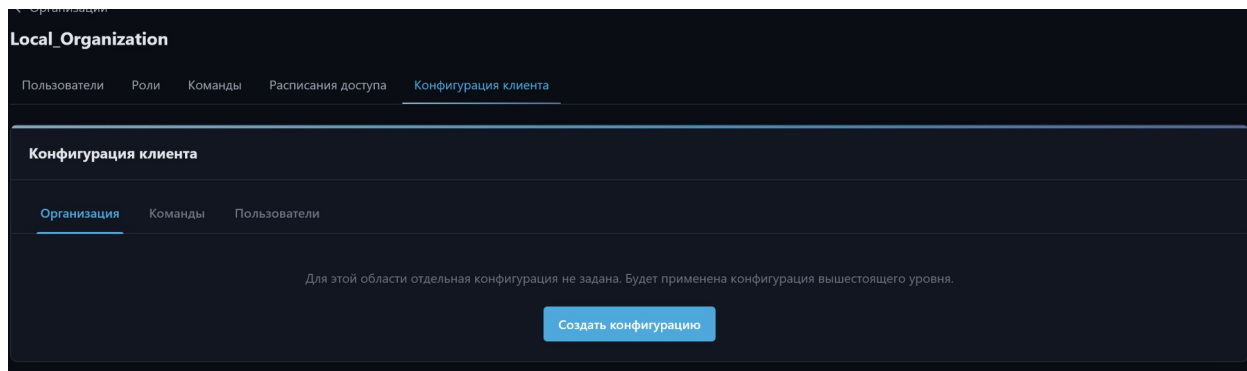


Рисунок 2 — Конфигурация клиента: область организации

4. Заполните форму по таблице. Звёздочкой в интерфейсе отмечены обязательные поля.

Блок	Поле	Что задать
«Провайдер»	«ID *»	Согласованный идентификатор поставщика
«Провайдер»	«Отображаемое имя *»	Название поставщика для отображения клиенту
«Провайдер»	«Базовый URL *»	Утверждённый базовый адрес подключения клиента
«Модели»	«Основная модель *»	Модель для основной работы клиента из списка доступных вариантов
«Модели»	«Модель сабagenta *»	Модель подчинённого агента
«Модели ролей»	Explorer	Модель для роли Explorer клиента; если отдельное назначение не требуется, оставьте «Не задана»
«Модели ролей»	Worker	Модель для роли Worker клиента; если отдельное назначение не требуется, оставьте «Не задана»
«Совместимость»	«Минимальная версия CLI»	Согласованную минимальную версию командного клиента; не принимайте показанный

Блок	Поле	Что задать
		пример за обязательное требование поставки

Конфигурация клиента

Организация Команды Пользователи

Провайдер

ID *

Отображаемое имя *

Базовый URL *

Модели

Основная модель *

Модель сабагента *

Модели ролей

Explorer

Worker

Совместимость

Минимальная версия CLI

Рисунок 3 — Форма конфигурации клиента

- Раскройте «Основная модель». В строке «Начните вводить для поиска» введите часть названия и выберите нужный вариант из списка. Затем выберите значение в поле «Модель сабагента». Используйте варианты текущей организации, а не названия из примеров или истории аналитики.
- Проверьте область настройки, адрес поставщика и все назначения моделей. Explorer и Worker — роли клиентского приложения, не роли администраторов ХАБ.
- Нажмите «Создать». Для выхода без создания используйте «Отмена».
- После создания повторно откройте ту же область конфигурации и проверьте сохранённые значения. Затем проверьте применение на целевом клиенте и выполните контрольный запрос по разделу 4.6. Сохранение в ХАБ само по себе не подтверждает получение конфигурации приложением.

3. Сравните копию с исходной конфигурацией: проверьте, что сохранённые параметры подходят новой модели.
4. Выполните предварительный просмотр спецификации. Проверьте модель, движок, источник весов и ссылки на секреты.
5. Исправьте ошибки проверки параметров до создания развёртывания. Некорректная спецификация отклоняется до обращения к кластеру; обходить проверку ручным созданием ресурсов не следует.

Предварительный просмотр не запускает модель и не резервирует ресурсы. После изменения сохранённой конфигурации отдельно проверьте параметры действующего развёртывания.

4.2. Развёртывание локальной модели

Локальная модель работает в вычислительной инфраструктуре платформенного контура. До запуска подготовьте конфигурацию, источник весов и доступ к нему, оцените возможность размещения.

1. Выберите подготовленную конфигурацию и проверьте её спецификацию.
2. Получите через продуктовый API перечень графических ускорителей (GPU) и предварительную оценку размещения. Оценка не является резервированием ресурсов.
3. Создайте развёртывание через предусмотренную контрактом операцию продуктового API. Серверная часть регистрирует ресурс модели через HTTP-адаптер.
4. Отслеживайте состояние созданного развёртывания. Оператор подготавливает кэш и том весов, создаёт экземпляры исполнения и отражает результат в состоянии ресурса.
5. После готовности выполните контроль доступа и запрос по разделу 4.6.

Принятие операции создания ещё не означает завершение запуска. При задержке или отказе используйте состояние, события и журналы именно созданного развёртывания.

Движки исполнения

Движок	Назначение
vLLM	Обслуживание языковых моделей
SGLang	Обслуживание языковых моделей
Ollama	Локальные модели и упрощённый контур
FasterWhisper	Обработка аудио и распознавание речи

Перед запуском проверьте совместимость модели и версии движка, требования к памяти и нужные виды запросов. Наличие аудиодвижка не означает поддержку всех мультимодальных функций.

Источники и кэш весов

Для загрузки весов используются HuggingFace, зеркала и S3-совместимые источники. Веса размещаются на постоянном томе: загрузчик подготавливает их во временном каталоге, затем устанавливает признак готовности. Вручную изменять каталоги для обхода этой последовательности не следует.

Если источник требует учётных данных, передайте их через секрет Kubernetes. В спецификации используйте ссылку на секрет, а не его значение. Не включайте ключи, пароли и токены в описания конфигураций, снимки экрана и отчёты.

При ошибке загрузки проверьте доступность источника, выбранную версию весов и диагностические сообщения через продуктовый API. Исправление доступа к хранилищу и обслуживание тома относятся к платформенному контуру.

Кэш весов хранит файлы модели для запуска. Учёт чтения и записи кэшированных токенов относится к обработке запросов и потреблению; это другой механизм.

GPU и размещение

Учитывайте ресурсы не только первоначального запуска, но и обновления. При замене экземпляра новая реплика запускается до удаления старой. Поэтому возможность разместить одну реплику не подтверждает достаточный запас для обновления. До изменения проверьте требования к памяти, число реплик и доступные ресурсы.

4.3. Управление состоянием развёртывания

Используйте сведения продуктового API для контроля нужного развёртывания.

Сведения	Назначение
Состояние	Проверить достижение требуемого состояния модели
События	Установить последовательность действий и причины задержки или отказа
Журналы	Разобрать ошибки загрузки и исполнения
Метрики	Оценить нагрузку, очередь, ошибки и использование GPU при включённой подсистеме наблюдаемости

После каждой операции проверьте принятие изменения, последующее состояние и контрольный запрос, если модель должна обслуживать обращения. Состояния задач машинного обучения не используйте как перечень состояний развёртывания. Отсутствие метрик само по себе не доказывает отказ: сбор метрик зависит от состава поставки.

Изменение и удаление

Перед изменением зафиксируйте исходную конфигурацию и перечень организаций, использующих модель. Передайте требуемое изменение через продуктовый контур и отслеживайте согласование состояния оператором.

При смене шаблона экземпляра исполнения оператор создаёт новую реплику и доводит её до готовности Ready, затем удаляет старую. Контролируйте наличие GPU для такой замены. После обновления проверьте состояние и контрольный запрос; не полагайтесь на автоматический возврат к прежней версии без предусмотренной процедуры восстановления.

Перед удалением согласуйте прекращение использования модели и судьбу постоянного тома, кэша весов и связанных секретов по правилам поставки. Удаление не следует использовать как временную остановку: сохранность связанных данных должна быть определена заранее.

Публикация и снятие с публикации

Доступность модели в продуктивном каталоге и существование развёртывания — разные состояния. Снятие с публикации не равнозначно удалению ресурса или освобождению GPU.

После изменения публикации проверьте состояние ресурса, видимость модели для целевой организации и возможность нового запроса. Если снятие с публикации не поддерживается, технический контракт предусматривает явный отказ, а не сообщение об успехе.

4.4. Масштабирование и расписание доступа

Механизмы масштабирования разделены:

- оператор выполняет переход между отсутствием работающих экземпляров и одним экземпляром (0↔1);
- KEDA изменяет число экземпляров от одного до нескольких (1↔N);
- Prometheus или VictoriaMetrics предоставляет метрики нагрузки.

Перед настройкой масштабирования проверьте наличие KEDA и источника метрик, поддерживаемые показатели нагрузки, пределы числа реплик и доступные вычислительные ресурсы. Пороговые значения задавайте по конфигурации выпуска и требованиям нагрузки.

При запуске из нулевого состояния возникает холодный старт. Запрос удерживается в пределах времени ожидания либо получает явный отказ HTTP 503. Учитывайте это при возобновлении работы модели.

Расписание доступа в организации

Для продуктовых окон доступа откройте «Организации» → карточку организации → «Расписания доступа». В блоке «Расписания доступа к моделям» нажмите «+ Создать расписание».

В форме «Новое расписание доступа» задаются «Область действия», «Модель», «Часовой пояс *» и флажок «Расписание включено». Модели для этого списка берутся из лимитов текущего тарифа организации. Если форма сообщает, что ни одной модели не задано, проверьте тариф; это не означает, что во всей системе отсутствуют модели.

В блоке «Еженедельные интервалы» выберите дни и время, при необходимости используйте «+ Добавить интервал» или «Убрать». Поля «Действует с» и «Действует по» задают даты действия; для исключений предусмотрена кнопка «+ Добавить исключение». Проверьте часовой пояс и интервалы перед нажатием «Создать». Не принимайте начальные значения формы за обязательные часы работы организации.

Расписание доступа ограничивает возможность запросов, а не управляет остановкой экземпляров. Закрытие продуктового окна не является командой освобождения GPU. Возобновление доступа не означает готовность модели.

4.5. Подключение внешнего поставщика

Режим `external` перенаправляет запросы к внешнему OpenAI-совместимому API. Продукт регистрирует внешнюю модель в каталоге, а платформа обращается к поставщику без создания локальных экземпляров исполнения этой модели. Работоспособность подключения отражается в состоянии.

Блок «Провайдер» в конфигурации клиента задаёт подключение приложения. Он не заменяет регистрацию внешней модели на стороне платформы.

1. Согласуйте внешнюю модель и нужные виды запросов, серверный адрес внешнего API, допустимость передачи данных и сетевой доступ к поставщику.

2. Подготовьте учётные данные в секрете Kubernetes и ссылку на него по правилам выпуска. Ключ внешнего поставщика не смешивайте с ключом запросов организации к ХАБ.
3. Зарегистрируйте внешнюю модель через предусмотренную контрактом операцию продуктового контура.
4. Проверьте запись модели и состояние подключения. Отсутствие локальных экземпляров исполнения в этом режиме нормально.
5. Проверьте разрешение модели целевой организации и выполните контрольный запрос через продуктовую точку входа ХАБ.

Проверяйте именно необходимый вид запроса, включая потоковую выдачу, если она нужна приложению: OpenAI-совместимость не гарантирует поддержку всех маршрутов и форматов данных. При частичном отказе регистрации сначала проверьте фактическое состояние записи и подключения, затем применяйте предусмотренный контрактом порядок восстановления или повторения операции.

4.6. Проверка доступа к модели

Готовность модели и право организации её использовать проверяются отдельно. До вызова серверная часть проверяет членство, подписку, разрешение модели, расписание и квоту. Для сервисного ключа применяются политики организации без привязки к пользователю и команде.

1. Определите организацию и вид доступа: пользовательский или сервисный. При пользовательском доступе проверьте членство в организации.
2. Откройте «Подписки», проверьте текущий тариф, срок действия и условия использования модели. Пометка «Без лимитов использования» не отменяет проверку подписки, прав и технической готовности.
3. Откройте «Квоты», в поле «МОДЕЛЬ» укажите публичное имя модели. Задайте нужное состояние флажка Streaming и нажмите «Показать». Проверьте результат для выбранной модели и режима. Сообщение «Для этой модели тариф не задаёт ограничений использования.» относится к тарифным ограничениям, а не доказывает работоспособность модели.
4. Просмотрите «Контракт учёта токенов». Статус «Не проверен» и значения «Не передаётся» не означают нулевое потребление. Не используйте отсутствующие значения категорий для вывода об остатке квоты или точной стоимости.
5. Проверьте применимое расписание. Для клиента дополнительно сверьте выбранного поставщика и модели с конфигурацией нужной области.
6. Направьте небольшой запрос с несекретными тестовыми данными через продуктовую точку `/api/v1/inference/{path}`. Разрешённый путь, идентификатор модели, способ аутентификации и формат тела возьмите из контракта выпуска. Прямой запрос к внутреннему шлюзу не заменяет эту проверку.
7. Убедитесь, что получен ответ выбранной модели. Откройте «Аналитика», задайте период и фильтр «МОДЕЛЬ», проверьте сведения о запросе и потреблении. При отказе используйте вкладку «Ошибки»: в таблице доступны дата, модель, пользователь, статус и сообщение. Нажатие на строку не открывает подробную карточку.

8. Зафиксируйте организацию, модель, вид доступа, время и результат проверки, а также идентификатор запроса, если он доступен. Не включайте ключи и чувствительное содержимое.

Если запрос отклонён до вызова модели, сначала проверьте продуктовые условия. При их выполнении продолжите разбор по состоянию локального развёртывания или внешнего подключения, событиям и журналам. Общее сообщение `Astra AI Hub inference request failed` не устанавливает первопричину отказа. Порядок диагностики приведён в главе 7.

Проверка завершена, когда подтверждены техническая работоспособность модели и её использование через ХАБ в нужной организации. Для централизованной настройки клиента дополнительно подтвердите фактическое применение сохранённых параметров в целевом приложении.

5. Безопасность

Глава определяет правила безопасного доступа к ХАБ, обращения с ключами, применения защиты содержимого и ведения аудита. Назначение ролей, управление пользователями и операции с ключами рассматриваются в главе 6; здесь приведены ограничения, которые необходимо сохранять при этих операциях.

Необязательные подсистемы не считаются включёнными по умолчанию. Доступ организуется по утверждённой сетевой схеме с разграничением полномочий и защищённым обращением с секретами. Прямой доступ к внутренним компонентам и отключение проверок не используются для устранения отказа доступа.

5.1. Аутентификация и доверенная личность

Аутентификация устанавливает личность пользователя или принадлежность сервисного ключа. Входную проверку выполняет APISIX; решение о праве на операцию принимает серверная часть ХАБ.

Для интерактивного входа используется OIDC с потоком Authorization Code и PKCE. APISIX проверяет токен. Выход завершает сессию системы идентификации; его нельзя считать отзывом всех видов доступа, включая сервисные ключи.

После входной проверки сведения о личности передаются доверенными заголовками. Клиентские заголовки семейства `X-User-*` не должны приниматься как доказательство личности; на входе они заменяются доверенными значениями. Аналогично клиент не должен получать возможность подменить внутренний ключ, который серверная сторона определяет для обращения к модели. Настройки промежуточного проксирования должны сохранять эту границу доверия.

Различайте два вида доступа:

- **Пользовательский доступ.** Пользователь проходит OIDC-аутентификацию. Для обращения к моделям предусмотрен персональный ключ. К запросу применяются политики организации, команды и пользователя.
- **Машинный доступ.** Сервисный ключ передаётся в `X-API-Key` на маршруте запросов. Он представляет организацию без пользователя и команды; к нему применяются политики организации. Не используйте такой ключ там, где требуется персональное разграничение по участникам или командам.

WebSocket консоли. Токен передаётся в подпротоколе `bearer,<token>`, проходит OIDC-проверку на APISIX и не передаётся далее как учётные данные. `<token>` обозначает

место токена. Полный заголовок подпротокола не включается в снимки, обращения и журналы. Защита WebSocket проверяется отдельно от обычного входа в консоль.

5.2. Авторизация и изоляция организаций

Организация — основная единица продуктовой изоляции: с ней связаны пользователи, роли, подписка и лимиты. Ей соответствует инфраструктурный арендатор (Tenant). Команда является группой внутри организации и областью применения политик; её нельзя считать отдельной организацией или самостоятельной инфраструктурной границей.

Полномочия на операции управления проверяет серверная часть. APISIX устанавливает доверенную личность, но не проверяет роли организации. Поэтому доступность страницы, скрытие элемента интерфейса или успешная аутентификация не заменяют серверную проверку прав.

Действуют следующие границы полномочий:

- роли создаются в рамках организации и состоят из разрешений фиксированного каталога;
- встроенная защищённая роль администратора организации даёт полный доступ в контуре этой организации;
- права администратора платформы отделены от ролей организации; через роль организации нельзя выдать платформенные полномочия на развёртывания, конфигурации моделей и связанные операции;
- администратору платформы не требуется обязательное членство в организации.

Проверка минимально необходимых прав

1. Откройте «Организации», выберите нужную карточку и во вкладке «Пользователи» найдите участника по EMAIL. Сверьте организацию, РОЛИ, КОМАНДА и СТАТУС.
2. Перейдите во вкладку «Роли» и найдите назначенную роль. Сопоставьте её название, описание и число прав.
3. Для изменяемой роли откройте «Редактировать», просмотрите отмеченные флажки «Права», затем нажмите «Отмена», если изменение не согласовано.
4. Сравните отмеченные разрешения с задачей участника. Название Viewer не означает просмотр всех экранов. Organization Admin отмечена как «системная роль»; в её строке отсутствуют кнопки редактирования и удаления. Эта роль относится к организации, а не ко всей платформе.
5. При недостатке доступа согласуйте конкретное разрешение и назначение, используя раздел 6.2. Не выдавайте платформенные полномочия как замену неясному праву организации.

Обозначения разрешений приведены в разделе 6.2. Для определения прав используйте состав роли, а не её название или число разрешений. Проверка флажков не заменяет проверку серверного отказа для запрещённой операции.

Перед вызовом модели серверная часть проверяет применимые условия: членство, активность подписки, разрешённую модель, расписание и квоту. Для сервисного ключа действует организационная область политик, а не персональные ограничения. При отсутствии активной подписки запросы к моделям и каталог моделей организации недоступны, но права управления организацией не снимаются. Эти состояния необходимо различать при разборе отказа.

При приёмке проверяют не только успешный доступ, но и отказ при обращении к объекту другой организации и при попытке выполнить платформенную операцию с организационными правами. Наличие сущности Tenant само по себе не подтверждает сетевую изоляцию, разделение всех хранилищ или невозможность доступа через иной маршрут.

5.3. Защита входных маршрутов и внутренних соединений

Продуктовая точка проксирования запросов — `/api/v1/inference/{path}`. Разрешённый запрос проходит продуктовые проверки до передачи в OpenAI-совместимый шлюз моделей. Поддержка шлюзом маршрута `/v1/*` не означает разрешения публиковать его для клиентов напрямую.

Входные и внутренние соединения сохраняют следующие ограничения:

1. Разрешены только зафиксированные пути; всё вне разрешённого перечня запрещено. Служебные интерфейсы шлюза наружу не публикуются.
2. Адрес назначения запроса задаётся на серверной стороне. Клиент не должен выбирать произвольный внутренний или внешний адрес через параметры запроса.
3. Консоль работает через продуктовый API. HTTP-адаптер вызывается серверной частью; консоль и внешние клиенты не получают прямого доступа к Kubernetes.
4. Внутренний заголовок с ключом доступа формируется после определения клиентского ключа. Он используется для служебного взаимодействия и не заменяет проверку доступа клиента.

Точный перечень опубликованных путей, методы запросов, порты и правила взаимодействия компонентов должны быть закреплены в сетевой схеме поставки. Не расширяйте разрешённый перечень по аналогии с возможностями OpenAI-совместимого интерфейса и не открывайте служебные точки ради устранения пользовательского отказа.

Сертификаты и сетевые ограничения

Проверяющий обработчик (webhook) использует TLS. Сертификаты и правила сетевого доступа обслуживаются по утверждённой сетевой конфигурации. При недоверенном сертификате устраняют причину нарушения доверия; проверку сертификата не отключают.

Сетевые правила разграничивают пользовательский доступ, внутренние связи и исходящие соединения. Успешная работа Kubernetes не заменяет контроль сетевой изоляции. Служебные интерфейсы остаются закрытыми для пользовательской сети.

Периметр доступа

VPN и средства периметра не заменяют проверки членства, подписки, моделей, расписаний и квот. Прямой маршрут к внутреннему шлюзу не используется для обхода продуктового прокси.

5.4. Защита ключей и секретов

Различайте назначение секретов: способ хранения одного вида не описывает автоматическое хранение остальных.

Вид	Назначение и обращение	Ограничение для эксплуатации
Персональный ключ запросов	Генерацию и учёт выполняет продукт; в	Не рассчитывайте на повторное получение

Вид	Назначение и обращение	Ограничение для эксплуатации
	кластер передаётся готовое значение; полный секрет показывается не более одного раза	полного значения; при утрате используйте предусмотренную смену ключа
Сервисный ключ организации	Машинный доступ без пользователя и команды; в базе хранится хеш секрета	Не приписывайте сервисному ключу пользовательские или командные ограничения
Учётные данные внешнего поставщика моделей	Хранятся в Kubernetes Secret	Не вставляйте значение секрета в описание модели, статус или диагностические материалы
Учётные данные источника весов	Передаются через Kubernetes Secret	Не размещайте их открытым текстом в спецификациях ресурсов и журналах загрузки

Выдача, смена и отзыв

Полный персональный секрет показывается не более одного раза. Смена делает старое значение недействительным, отзыв отключает клиента API. После изменения обновляется кэш определителя ключей. Секрет не попадает в журналы и ответы кластерного API.

После смены проверьте отказ старого значения и согласованность состояния на экземплярах, использующих кэш. Создание нового ключа само по себе не подтверждает прекращение старого доступа. Операции с персональными и сервисными ключами выполняются отдельно.

Для выдачи назначьте согласованный защищённый способ передачи, исключаящий общие чаты, открытые заявки и снимки экрана с полным значением. При подозрении на компрометацию действуйте по утверждённому регламенту: определите вид и владельца секрета, ограничьте скомпрометированный доступ предусмотренной для него операцией, согласуйте замену у зависимых систем и проверьте прекращение старого доступа. Для секрета внешнего поставщика требуется согласование и на стороне поставщика; смена пользовательского ключа ХАБ его не заменяет. Конкретные операции управления ключами относятся к главе 6 и документации выпуска.

Безопасная проверка сервисных ключей в панели

1. Откройте «Сервисные API-ключи» и сопоставьте название, дату создания, статус и дату отзыва с разрешёнными интеграциями. Не копируйте содержимое колонки КЛЮЧ.
2. Проверьте счётчик: интерфейс учитывает только активные ключи. Сверьте разрешённый предел с разделом «Подписки», а не с числом в примере.
3. Для нового ключа заранее определите владельца интеграции и защищённый способ передачи. В форме создания укажите только назначение в «Название *»; не помещайте туда секрет или пароль.

4. При прекращении работы интеграции используйте порядок отзыва в разделе 6.4. После действия проверьте запись в списке и отдельно согласованную проверку прекращения доступа.
5. Перед отправкой снимка, загрузки или обращения проверьте, что значения ключей и данные посторонних участников скрыты. Полный секрет не нужен для сопоставления интеграции с записью.

Форма создания сервисного ключа содержит поле «Название *». Название не задаёт срок действия или разрешения ключа. Правило однократного показа персонального ключа не используется как правило хранения сервисного секрета.

Хранение и передача

В инфраструктурном контуре для хеширования ключей используется `argon2id`; сервисные ключи хранятся в базе в виде хеша. Хеширование ключей не является шифрованием всех сохраняемых данных.

Секреты внешних систем передаются через ссылки на Kubernetes Secret, а не открытыми значениями в спецификации или статусе. Kubernetes Secret не заменяет ограничения прав чтения, защиту хранилища и резервных копий.

Права чтения, изменения и удаления секретов предоставляются только уполномоченным учётным записям. Локальное хранилище AstraCode не является хранилищем секретов ХАБ.

5.5. Защита содержимого на шлюзе

Guardrails — отдельный включаемый слой шлюза моделей. По умолчанию защита содержимого выключена и не изменяет запросы и ответы. Наличие шлюза не означает включение защиты.

Политики задаются общекластерным ресурсом `GuardrailPolicy` и связываются с моделью и/или клиентом. Оператор формирует фактически применяемый набор и отражает его в статусе модели; шлюз использует этот набор без прямого доступа к общекластерным политикам. Поэтому наличие созданной политики ещё не подтверждает её применение к нужному запросу: требуется проверить привязку и итоговый набор.

Различайте проверку до вызова модели (`pre_call`), перед проксированием (`during_call`), проверку ответа, включая SSE (`post_call`), и режим только аудита (`logging_only`). Блокировка (`block`), маскирование (`mask`) и аудит (`audit`) имеют разные последствия. Режим только аудита не блокирует передачу данных.

Типы проверок

Тип детектора	Предмет проверки	Что необходимо учитывать
<code>pii</code>	Персональные данные	Маскирование не предоставляет разрешение на внешнюю передачу данных
<code>secret</code>	Ключи и токены	Наличие детектора не даёт гарантии обнаружения любого секрета
<code>denylist</code>	Запрещённые формулировки	Применяется согласованный перечень;

Тип детектора	Предмет проверки	Что необходимо учитывать
		охват и ложные срабатывания проверяются на контрольных данных
<code>prompt_injection</code>	Попытки подмены инструкций и обхода ограничений модели	Используются эвристики, сигнатуры и порог; типовые действия — блокировка или аудит, а не маскирование
<code>external</code>	Проверка внешним HTTP-сервисом	Адрес задаётся конфигурацией платформы, не клиентским запросом; передачу содержимого этому сервису необходимо согласовать

Обязательные политики и отказ проверяющего сервиса

Политики с признаком `defaultOn` обязательны: клиент не может отключить их параметрами запроса. Попытка снять обязательную политику отклоняется. Блокировка содержимого отражается как отказ политики безопасности, а не как ошибка модели.

Обязательность конкретной политики и включённость всей подсистемы — разные условия. Признак `defaultOn` не является подтверждением работы защиты при выключенном слое Guardrails.

Отказ детектора по умолчанию обрабатывается в закрытом режиме: проверяемое содержимое не пропускается. Открытый режим допускается только явной конфигурацией и оставляет наблюдаемый след. Его не включают как способ устранения недоступности модели без разрешённого решения по безопасности.

Срабатывания политик фиксируются в аудите и метриках без тел запросов и ответов. Для проверки используйте согласованные искусственные данные, а не реальные персональные данные или действующие ключи.

Ограничения защиты

Эвристический детектор внедрения инструкций дополняет разграничение доступа, ограничение маршрутов и аудит, но не заменяет их. Отсутствие срабатывания не доказывает безопасность текста, а ложное срабатывание требует разбора политики, а не автоматического отключения защиты.

Проверка ответа не предотвращает передачу исходного запроса модели. Для потоковой выдачи уже отправленный фрагмент не отзывается проверкой последующего фрагмента. Наличие проверки SSE не означает гарантии отсутствия утечки.

Текстовые детекторы не используются как универсальная защита изображений, аудио, вложений и вызовов инструментов.

5.6. Аудит и защита сохраняемых данных

Журнал аудита фиксирует, кто выполнил действие, что произошло, когда и с каким результатом. Он не содержит ключей, тел запросов и текстов обращений к модели. Сквозной идентификатор запроса позволяет сопоставить аудит, учёт и материалы инцидента.

Не смешивайте следующие виды данных:

- **Аудит безопасности:** действия и результаты без содержимого запросов и секретов.
- **Учёт и технические метрики:** потребление, статус, задержки и связанные технические сведения.
- **Содержимое диалогов:** тексты и иные данные запросов и ответов, сбор которых допускается только по соответствующей политике.

Сбор технических событий не предоставляет разрешения сохранять содержимое диалогов. До включения сохранения содержимого закрепляются категории данных, цель, хранилища и получатели, права, сроки хранения, порядок удаления и обращения с резервными копиями. Без этих условий сохранение для диагностики не включают.

Продуктовый журнал и платформенные данные используют разные схемы и роли PostgreSQL. Доступ к аналитике использования не предоставляет права чтения аудита или диалогов.

Подготовка безопасного фрагмента аналитики

1. В «Аналитика» ограничьте период и модель нужным случаем, при необходимости выберите команду.
2. Откройте «Ошибки», найдите запись и сохраните дату со временем и часовым поясом, модель, пользователя в допустимом объёме, статус и текст ошибки. Подробности по нажатию строки не открываются; в таблице отсутствует идентификатор запроса.
3. При наличии идентификатора в разрешённом аудите или журнале добавьте его с указанием источника. Если его нет, передайте сочетание времени, модели и пользователя для дальнейшего сопоставления, не выдавая его за уникальный системный идентификатор.
4. Проверьте текст и вложения: исключите ключи, токены, заголовки аутентификации и содержимое запросов и ответов. Для CSV отдельно проверьте строки посторонних участников и лишний период.

Если требуется разбор сохранённого диалога, используйте отдельно согласованный доступ и регламент, а не копирование текста в общую заявку. Порядок диагностики по таблице ошибок приведён в разделе 7.1.

5.7. Границы гарантий безопасности

Безопасность ХАБ зависит от продуктовых проверок и инфраструктурных настроек. За содержание политик, их применение оператором и шлюзом, инфраструктуру и разбор отказов назначаются ответственные по соответствующим функциям.

Работа в кластере заказчика не означает, что данные никогда не покидают его контур. Передача возможна при использовании внешних моделей и внешнего проверяющего сервиса; состав событий и получателей также зависит от включённых функций. Разрешение на такую передачу, категории допустимых данных и ограничения исходящих соединений должны быть согласованы до использования соответствующего маршрута. Маскирование не

следует считать разрешением на внешнюю передачу без отдельного решения владельца данных.

При допуске к эксплуатации контролируются:

- версия и состав поставки, применимая модель угроз и утверждённая топология без обхода продуктовых проверок;
- серверное разграничение прав, изоляция организаций и защита всех входных каналов;
- владельцы и жизненный цикл ключей, секретов и сертификатов;
- включённость, область действия и режимы отказа политик защиты содержимого;
- правила внешней передачи, аудита, хранения, удаления и допустимой диагностики;
- порядок реагирования на компрометацию и распределение ответственности между эксплуатационными командами.

Размещение в собственном кластере и наличие детекторов не являются доказательством сертификации, соответствия нормативным требованиям, абсолютной изоляции или обнаружения всех атак. Применение мер защиты оценивается в рамках модели угроз и фактической конфигурации.

6. Администрирование

Администрирование ХАБ охватывает управление организациями и доступом, контроль потребления, наблюдение за работой системы и сопровождение её конфигурации и данных. В продуктовом контуре ведутся организации, пользователи, команды, роли, подписки, ключи и политики. Платформенный контур обеспечивает работу моделей и вычислительной инфраструктуры. Изменения, затрагивающие оба контура, требуют согласованного состояния продуктовых записей и инфраструктурных ресурсов.

После изменения повторно откройте соответствующий список или форму и проверьте результат. Успешная отправка формы не заменяет проверку сохранённого состояния и фактического доступа.

6.1. Организации, пользователи и команды

Организация и инфраструктурный арендатор

Организация — единица изоляции продуктового контура. С ней связаны пользователи, роли, подписка и лимиты. Команда является группой внутри организации, а не отдельной организацией и не инфраструктурным арендатором.

Инфраструктурное представление организации — ресурс `Tenant` с именем вида `hub-org-<uuid>`. Он относится к платформенному контуру. Для пользовательского доступа предусмотрен клиент API с именем, начинающимся с `hub-user-`, для машинного доступа организации — с `hub-service-`. Эти ресурсы не заменяют продуктовые сущности: наличие арендатора само по себе не подтверждает наличие активной подписки или разрешённого пользовательского доступа.

Организация связана с инфраструктурным арендатором. HTTP-адаптер не генерирует секреты организации при регистрации арендатора. Ручное создание дополнительного `Tenant` не используется для исправления частично выполненной продуктовой операции.

При сопровождении организации значимы её состав, связь с арендатором, действующая подписка, применяемые роли и политики. Сведения об этих объектах следует рассматривать

совместно: успешное изменение только одной стороны не доказывает завершённость изменения в целом.

Открытие организации и проверка состава

1. В боковом меню выберите «Организации».
2. Найдите нужную карточку. Сверьте название и идентификатор организации, статус, число пользователей и дату создания.
3. Нажмите карточку. На странице организации доступны вкладки «Пользователи», «Роли», «Команды», «Расписания доступа», «Конфигурация клиента».
4. Откройте «Пользователи». В таблице «Пользователи организации» сопоставьте колонки EMAIL, РОЛИ, КОМАНДА и СТАТУС. Прочерк в колонке ролей не используйте как доказательство отсутствия любых полномочий.
5. Для перехода к другой организации используйте ссылку «Организации» над названием и заново сверьте карточку. Не переносите выводы о составе и правах между организациями.

Наличие ресурса Tenant не заменяет регистрацию организации в продуктовом контуре.

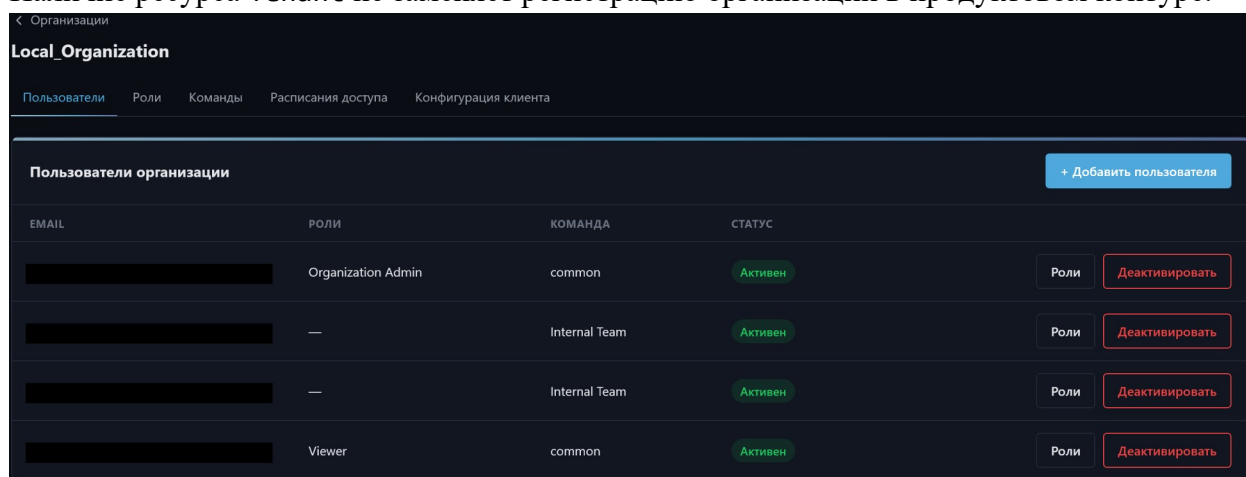


Рисунок 5 — Пользователи организации и доступные действия

Добавление пользователя

1. Откройте «Организации» → нужная организация → «Пользователи».
2. Нажмите «+ Добавить пользователя».
3. В окне «Добавить пользователя» заполните обязательное поле «Email *» адресом согласованного участника.
4. В группе «Роли» отметьте согласованную роль, предварительно проверив её разрешения по разделу 6.2. В группе «Команды» отметьте нужную доступную команду. Не выбирайте расширенные права только ради завершения формы.
5. Сверьте адрес и выбор, затем нажмите «Добавить». Чтобы выйти без отправки формы, нажмите «Отмена» либо значок закрытия.
6. Повторно откройте список пользователей и проверьте наличие участника, его роли, команду и статус. При отсутствии записи или сообщении об ошибке зафиксируйте результат до повторной отправки.

Форма добавления пользователя не запрашивает пароль.

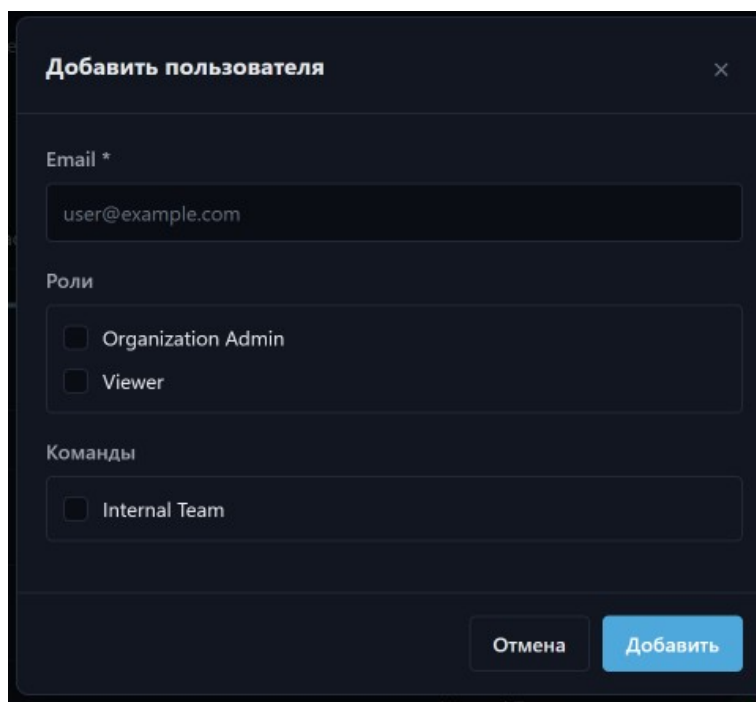


Рисунок 6 — Добавление пользователя

Действия с существующим участником

В строке активного пользователя доступны «Роли» и «Деактивировать». Для проверки назначения найдите участника по EMAIL и сопоставьте текущие значения в колонке РОЛИ. Кнопка «Роли» относится к назначениям участнику, а «Редактировать» во вкладке «Роли» — к составу разрешений самой роли; это разные изменения.

Перед деактивацией согласуйте прекращение доступа и проверьте, не затрагивается ли единственный администратор организации. Если действие разрешено, используйте «Деактивировать» только в строке выбранного участника и затем перечитайте её статус. Изменение статуса не доказывает немедленный отзыв всех ключей, завершение текущих сеансов или удаление истории; эти последствия определяются отдельным регламентом.

Членство пользователя

У пользователя допускается не больше одной активной привязки к организации. Состав организации синхронизируется с системой идентификации DEX (Keycloak). Вход пользователя и его членство — разные условия: подтверждённая личность не заменяет проверку организационной принадлежности при запросе к модели.

Удаление учётной записи в системе идентификации не используется как замена управлению членством в ХАБ. При прекращении доступа отдельно контролируются членство, ключи и активные интеграции. Изменение членства не является операцией удаления истории запросов.

Команды

Команда группирует пользователей внутри организации и служит областью действия политик. В пользовательском запросе применяются политики организации, команды и самого пользователя.

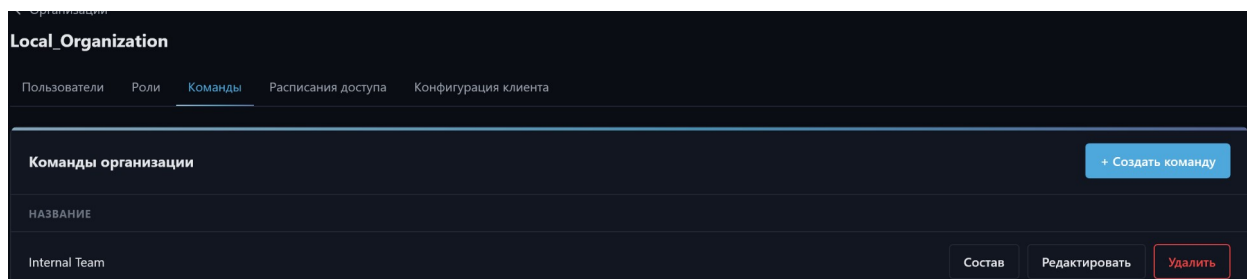


Рисунок 7 — Список команд организации

Создание и переименование команды

1. Откройте организацию → «Команды». В блоке «Команды организации» нажмите «+ Создать команду».
2. В окне «Новая команда» заполните «Название *» и нажмите «Создать»; для выхода без изменения — «Отмена».
3. Повторно откройте список и найдите команду по названию.
4. Для переименования нажмите «Редактировать» в строке нужной команды. В окне «Редактировать команду» измените «Название *» и нажмите «Сохранить».
5. Проверьте название в списке.

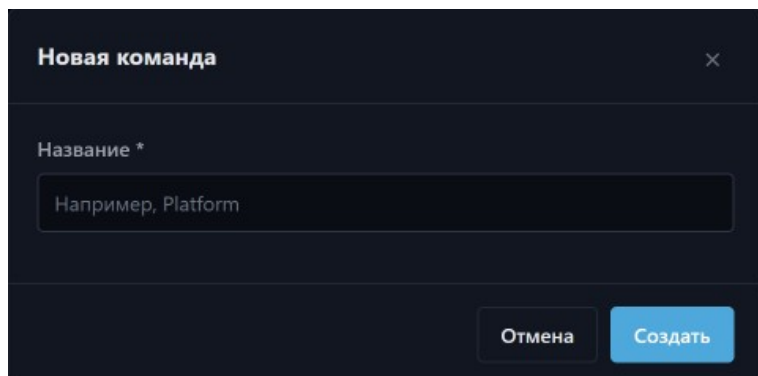


Рисунок 8 — Создание команды

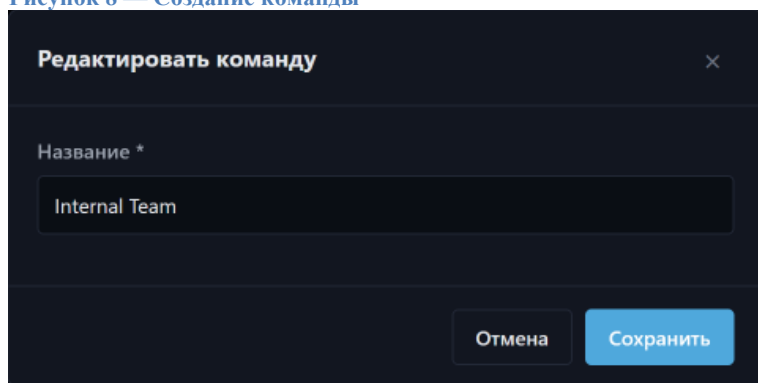


Рисунок 9 — Редактирование команды

Добавление и исключение участников команды

1. Во вкладке «Команды» найдите нужную строку и нажмите «Состав».
2. Сверьте название команды в заголовке «Состав команды: ...».
3. Для включения участника найдите его в блоке «ДОБАВИТЬ УЧАСТНИКА» и нажмите «Добавить» именно в его строке.
4. Для исключения найдите участника в блоке «УЧАСТНИКИ» и нажмите «Убрать» в его строке.

5. Повторно откройте «Состав» и проверьте принадлежность изменённого участника. Сопоставьте также колонку КОМАНДА во вкладке «Пользователи».

«Убрать» относится к составу команды, а не к удалению учётной записи из организации. Перед удалением команды проверьте связанные политики и участников. Изменение команды не ограничивает сервисный ключ организации. Обновление списка участников не заменяет проверку применения доступа.

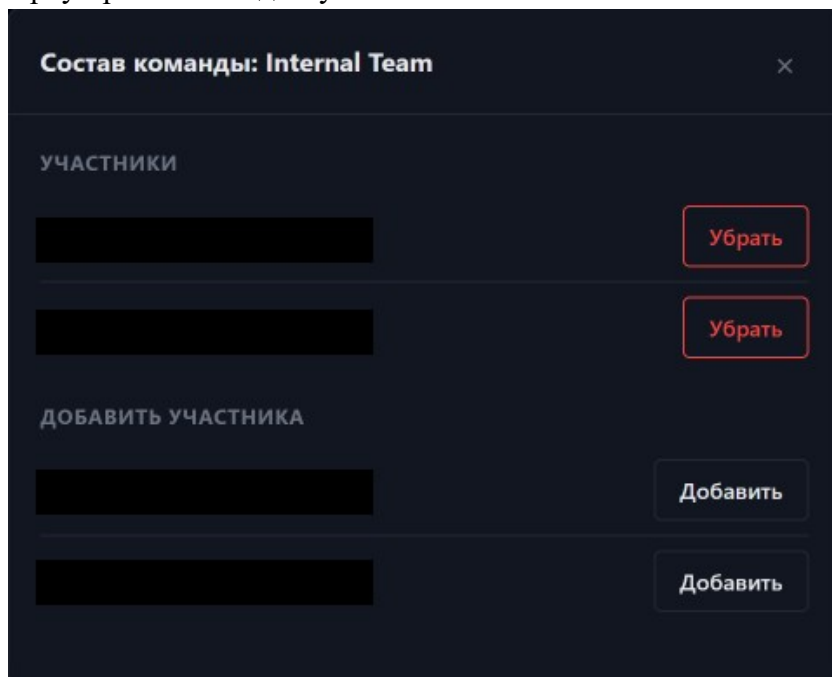


Рисунок 10 — Состав команды: добавление и исключение участников

6.2. Роли и разрешения

Роли организации

Роли создаются в пределах организации и состоят из разрешений фиксированного каталога. Произвольное название роли не создаёт нового разрешения. Решение о допустимости операции управления принимает серверная часть ХАБ. APISIX подтверждает личность и формирует доверенные заголовки, но не проверяет роли организации.

Встроенная защищённая роль администратора организации даёт полный доступ в пределах организации. Это не роль администратора всей платформы. Перед изменением состава администраторов сохраняйте доступ хотя бы одного уполномоченного администратора организации.

При проверке полномочий необходимо различать:

- организацию, в которой действует роль;
- разрешения, входящие в роль;
- фактическое назначение роли пользователю;
- ограничения подписки и политик на запросы к моделям.

Наличие административного разрешения не отменяет условия пользовательского запроса. В частности, отсутствие активной подписки блокирует запросы к моделям, но не снимает права управления организацией.

Проверка разрешений роли

1. Откройте организацию → «Роли». Таблица «Роли организации» содержит название, описание и число прав.
2. Для изменяемой роли нажмите «Редактировать» и просмотрите отмеченные флажки «Права», включая прокручиваемую часть списка. Для проверки без изменения нажмите «Отмена».
3. Сопоставьте набор с задачами участника, а не только с названием или описанием роли.

Organization Admin помечена как «системная роль»; в её строке отсутствуют кнопки редактирования и удаления. Права изменяемых ролей определяются отмеченными разрешениями. Название Viewer не задаёт универсальный доступ на чтение.

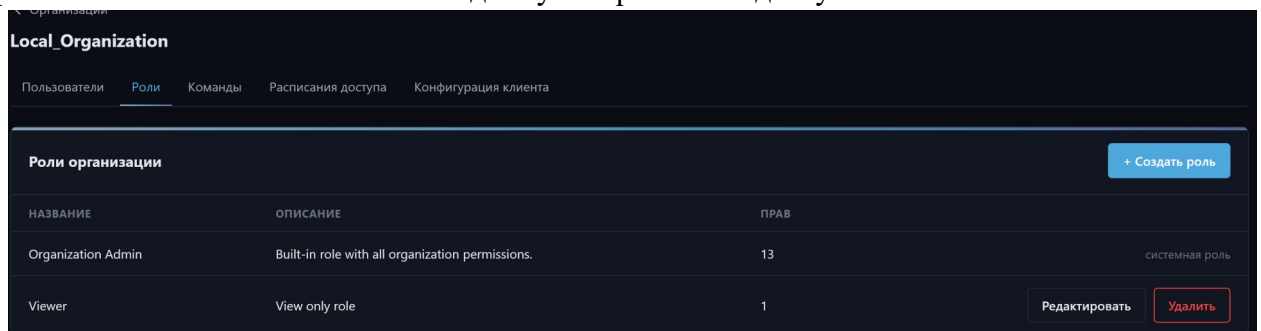


Рисунок 11 — Роли организации

Обозначения отдельных разрешений:

Разрешение	Назначение
usage_stats:view:own	Просмотр собственной статистики использования
usage_stats:view:organization	Просмотр статистики организации
teams:view	Просмотр команд
teams:manage	Управление командами
teams:members:view	Просмотр состава команд
teams:members:manage	Управление составом команд
organization:members:view	Просмотр участников организации
organization:members:manage	Управление участниками организации
organization:member_roles:manage	Управление назначением ролей участникам организации

Для определения полного набора прав просмотрите весь прокручиваемый список в форме роли.

Редактировать роль

Название *

Viewer

Описание

View only role

Права

- ☐ usage_stats:view:own
- ☐ usage_stats:view:organization
- ☒ teams:view
- ☐ teams:manage
- ☐ teams:members:view
- ☐ teams:members:manage
- ☐ organization:members:view
- ☐ organization:members:manage
- ☐ organization:member_roles:manage

Отмена Сохранить

Рисунок 12 — Просмотр разрешений изменяемой роли

Создание и редактирование роли

1. Во вкладке «Роли» нажмите «+ Создать роль».
2. В окне «Новая роль» заполните «Название *», при необходимости — «Описание».
3. В списке «Права» отметьте только согласованные разрешения. Просмотрите весь доступный список; не ограничивайтесь первым видимым фрагментом.
4. Нажмите «Создать» либо «Отмена» для выхода без отправки. Затем проверьте название, описание и число прав в таблице, повторно откройте форму и сверьте флажки.
5. Для изменения существующей роли нажмите «Редактировать» в её строке. В окне «Редактировать роль» скорректируйте название, описание или флажки и нажмите «Сохранить». Повторно сверьте набор.
6. Отдельно проверьте назначение роли участникам во вкладке «Пользователи»: создание роли само по себе не назначает её пользователю.

Перед изменением используемой роли определите затрагиваемых участников: изменение общего набора не является выдачей права только одному человеку. Кнопка «Удалить» относится к самой роли; не используйте удаление для проверки её назначений.

Новая роль

✕

Название *

Например, Team Viewer

Описание

Необязательно

Права

☐ usage_stats:view:own

☐ usage_stats:view:organization

☐ teams:view

☐ teams:manage

☐ teams:members:view

☐ teams:members:manage

☐ organization:members:view

☐ organization:members:manage

☐ organization:member_roles:manage

Отмена

Создать

Рисунок 13 — Создание роли и выбор разрешений

Платформенное администрирование

Права на развёртывания, конфигурации моделей и связанные платформенные операции доступны администратору платформы. Их нельзя выдать через роль организации. Для платформенного администратора членство в организации не является обязательным.

Назначение платформенных прав выполняется отдельно от изменения ролей организации. Произвольное назначение групп в системе идентификации или ручная правка базы данных не заменяют управление правами ХАБ.

6.3. Тарифы и подписки

Каталог тарифов

Каталог тарифов ведёт администратор платформы. Тариф задаёт ограничения и условия использования для организации:

Параметр	Назначение
Разрешённые модели	Определяет перечень моделей, доступных по тарифу
Предел числа пользователей	Ограничивает состав организации по правилам тарифа

Параметр	Назначение
Предел числа сервисных ключей	Ограничивает количество машинных ключей организации
Квоты потребления	Задают объём использования, в том числе по моделям и категориям токенов
Цена периода	Задаёт стоимость периода в продуктовом учёте

Тарифные ограничения и цена периода относятся к продуктовому учёту. Базовый контур тарифов, подписок и квот не проводит внешнюю оплату. Наличие цены не означает оформление платёжных или расчётных документов.

Просмотр текущей подписки и сравнение тарифов

1. Выберите «Подписки» в боковом меню либо перейдите по ссылке «К подписке» из раздела «Квоты».
2. В текущей подписке проверьте название тарифа, начало и срок действия, лимит сервисных API-ключей.
3. В карточках доступных тарифов сопоставьте ограничения пользователей и сервисных ключей, модельные лимиты, цену, валюту и период; описание учитывайте при его наличии.
4. Текущую карточку определяйте по пометкам «Текущий» и «Уже активен». Надпись «Без лимитов использования» относится к условиям тарифа: она не отменяет права, расписания, ограничения инфраструктуры и доступность самой модели.
5. У доступного тарифа кнопка «Оформить» служит для перехода к оформлению; она не используется для проверки доступности модели.
6. После оформления повторно откройте «Подписки» и сверьте текущий тариф, даты и лимиты. Нажатие «Оформить» само по себе не означает активацию или оплату.

Отсутствие перечня модельных лимитов не доказывает отсутствие моделей или доступа. Каталог и выбор модели в конфигурации клиента рассматриваются в главе 4, отдельно от истории аналитики.

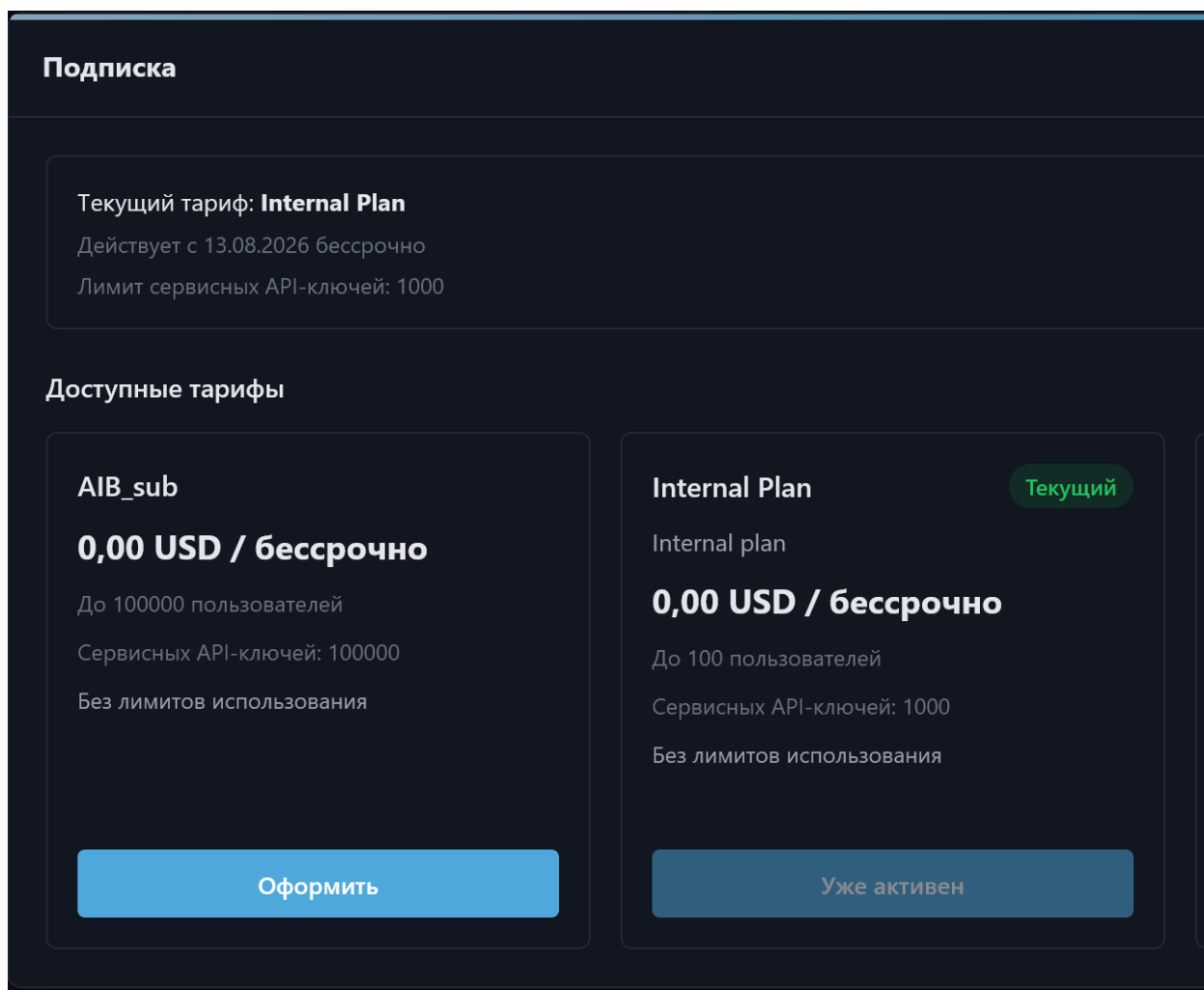


Рисунок 14 — Текущая подписка и карточки тарифов

Состояние подписки

Без активной подписки организации недоступны запросы к моделям и каталог моделей организации, в том числе каталог для конфигурации клиента. Права управления организацией при этом сохраняются. Следовательно, доступность административной части не доказывает, что организация может выполнять запросы.

Для доступа организации требуется согласованное состояние подписки, инфраструктурного арендатора и пользовательских ключей активных участников. Частичное завершение операции не является завершённой настройкой. При расхождении сохраните сведения об объектах и передайте инцидент по главе 7; не создавайте дополнительные ресурсы и ключи вручную.

Лицензирование поставки

Тариф и подписка регулируют потребление внутри ХАБ. Право использования программного обеспечения и условия сопровождения оформляются отдельно. Процедуры активации лицензии AstraCode не применяются к ХАБ.

6.4. Пользовательские и сервисные ключи

Персональный доступ

Пользовательский ключ предназначен для запросов конкретного пользователя к моделям. Его генерирует и учитывает продуктовый контур; в кластер передаётся готовое

значение. Формат задан как `sk-<8hex>-<32hex>`; это обозначение структуры, не пример действующего секрета.

Клиент пользователя связан с инфраструктурным арендатором, и эта привязка неизменяема. Поэтому перевод пользователя в другую организацию нельзя описывать как простую замену арендатора у существующего клиента. Соответствующий порядок должен учитывать членство, ключи и историю использования.

Для пользовательского ключа установлены следующие свойства:

- полный секрет предоставляется не более одного раза;
- смена ключа делает прежнее значение недействительным;
- отзыв отключает клиента;
- после изменений обновляется кэш определения ключей;
- секрет не должен попадать в журналы и ответы кластерного API.

Выдача, смена и отзыв — разные изменения состояния. При смене обновляют значение в использующем его приложении и проверяют отказ старого ключа. Отзыв проверяют по прекращению доступа, а не только по состоянию записи.

Определение ключей на входе использует ValKey и запасное чтение PostgreSQL. Само наличие нового значения в одной системе недостаточно для вывода о завершении смены: результат должен соответствовать состоянию клиента и кэша определения ключей.

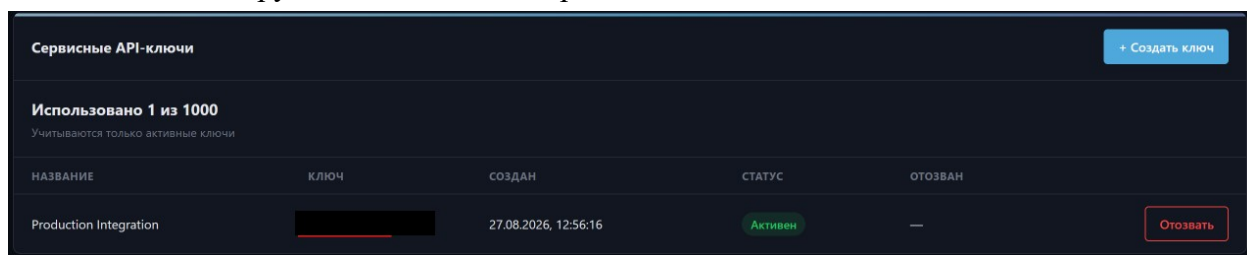
Машинный доступ организации

Сервисный ключ предоставляет машинный доступ от имени организации без пользователя и команды. Для его запросов действуют только политики организации. Персональное расписание или командная квота не распространяются на него по аналогии с пользовательским ключом.

В базе хранится хеш сервисного секрета. Число активных ключей ограничено тарифом; счётчик сопровождается пояснением «Учитываются только активные ключи». Персональные и сервисные ключи учитываются и обслуживаются отдельно.

Просмотр и создание сервисного ключа

1. Выберите «Сервисные API-ключи» в боковом меню.
2. Проверьте счётчик «Использовано ... из ...». При достижении предела сначала согласуйте состав интеграций и условия тарифа, не создавайте несколько ключей параллельно для обхода лимита.
3. В таблице сопоставьте НАЗВАНИЕ, СОЗДАН, СТАТУС и ОТОЗВАН. Колонку КЛЮЧ не копируйте в отчёты или обращения.



Сервисные API-ключи					+ Создать ключ
Использовано 1 из 1000					
Учитываются только активные ключи					
НАЗВАНИЕ	КЛЮЧ	СОЗДАН	СТАТУС	ОТОЗВАН	
Production Integration		27.08.2026, 12:56:16	Активен	—	Отозвать

Рисунок 15 — Список сервисных API-ключей

4. Нажмите «+ Создать ключ». В окне «Создать сервисный API-ключ» заполните «Название *» так, чтобы было понятно назначение интеграции, без секретов и паролей.

5. Если поле пустое и показано «Обязательно для заполнения», введите название.
6. Нажмите «Создать»; для выхода без отправки используйте «Отмена» или значок закрытия. При предоставлении секрета передайте его защищённым способом по разделу 5.4.
7. Повторно откройте список, проверьте запись по названию, статус и счётчик активных ключей. Если исход операции неизвестен, сначала установите наличие записи, а не отправляйте создание повторно.

Рисунок 16 — Создание сервисного API-ключа: обязательное название

Отзыв сервисного ключа

1. Определите интеграцию и согласуйте прекращение её доступа или замену ключа.
2. В «Сервисные API-ключи» найдите нужную активную запись по названию и дате создания, сверьте владельца интеграции без обмена значением секрета.
3. Нажмите «Отозвать» в нужной строке. Перед подтверждением действия сверяйте выбранную запись.
4. Повторно откройте таблицу и проверьте СТАТУС, ОТОЗВАН и счётчик. Прекращение доступа интеграции проверяйте отдельно по согласованному сценарию; одно обновление строки не доказывает завершение текущих запросов.

Для сопровождения машинного доступа важны организация, назначение интеграции, число учитываемых тарифом ключей и подтверждённое состояние каждого клиента. Секреты не включаются в руководство, снимки экрана, отчёты о проверках и журналы. Утрата полного значения не является основанием для извлечения его из базы данных или обхода ограничений показа.

6.5. Квоты, лимиты и расписания доступа

Области действия ограничений

До вызова модели продукт проверяет членство, активную подписку, разрешённую модель, расписание и квоту. Отказ политики не приводит к вызову модели. Для пользователя применяются уровни организации, команды и пользователя; для сервисного ключа — уровень организации. Различайте область каждого ограничения: показанная квота одной области не отменяет ограничений другой.

Просмотр квот выбранной модели

1. Откройте «Квоты» в боковом меню. На странице «Квоты использования» найдите поле «МОДЕЛЬ» с подсказкой «Публичное имя модели».
2. Введите публичное имя нужной модели. Подсказки строятся по использованию за 90 дней: отсутствие подсказки не является доказательством запрета или отсутствия модели.
3. Установите флажок Streaming для потокового режима либо снимите его для обычного ответа и нажмите «Показать».

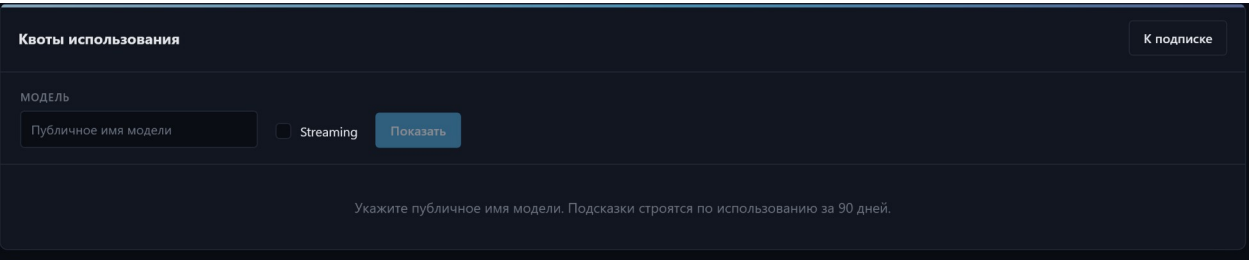


Рисунок 17 — Квоты: выбор модели и режима ответа

4. Прочитайте результат именно для выбранного имени и режима. Сообщение «Для этой модели тариф не задаёт ограничений использования.» означает отсутствие тарифного ограничения в показанном результате, а не отсутствие всех ограничений и не проверку доступности модели.
5. Просмотрите «Контракт учёта токенов» по инструкции ниже. При сравнении режимов измените Streaming и снова нажмите «Показать»; одинаковые условия одного имени не переносите на другие модели.
6. Зафиксируйте время последнего обновления. Экран содержит подпись «Автообновление каждые 10 секунд»; при подозрении на устаревшие сведения проверяйте фактическое время, а не только наличие подписи.
7. Для сопоставления с условиями тарифа нажмите «К подписке».

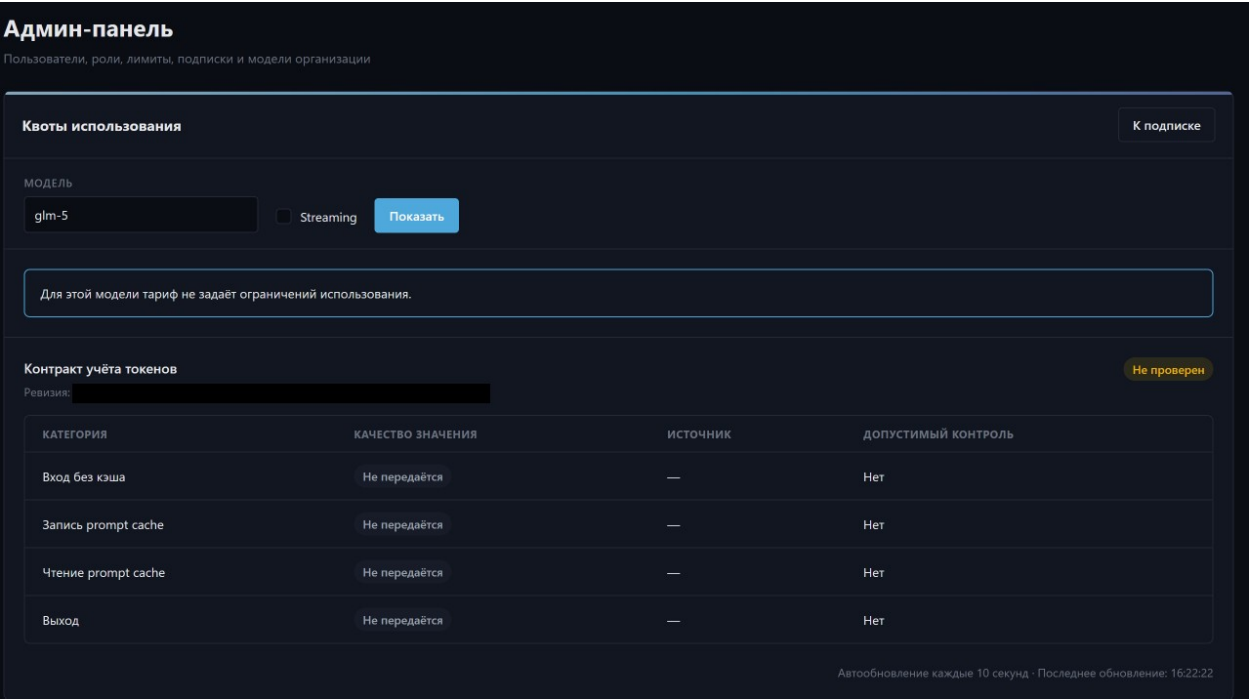


Рисунок 18 — Результат просмотра квот и контракт учёта токенов

Чтение контракта учёта токенов

Контракт описывает представление категорий токенов для выбранной модели и режима; его нельзя подменять общим итогом потребления организации.

1. В блоке «Контракт учёта токенов» запишите ревизию и статус. «Не проверен» не означает подтверждённый точный учёт.
2. Для каждой категории сопоставьте колонки «КАЧЕСТВО ЗНАЧЕНИЯ», «ИСТОЧНИК», «ДОПУСТИМЫЙ КОНТРОЛЬ»:

Категория на экране	Что сопоставить
«Вход без кэша»	Сведения об обычных входных токенах
«Запись prompt cache»	Сведения о записи кэша входного запроса
«Чтение prompt cache»	Сведения о чтении кэша входного запроса
«Выход»	Сведения о выходных токенах

3. Если указано «Не передаётся», источник обозначен прочерком, а допустимый контроль — «Нет», сохраните эти значения как отсутствие передаваемых сведений и указанного допустимого контроля по категории. Не записывайте ноль и не делайте вывод об отсутствии учёта во всей системе.
4. При расхождении с аналитикой приложите модель, режим, ревизию, статус, значения категорий и время обновления к диагностике по главе 7. Не исправляйте контракт или журнал вручную ради совпадения итогов.

Потребление и интенсивность запросов

Различайте объём потребления и интенсивность запросов. Категории кэшированных токенов в контракте учёта не равнозначны кэшу весов модели или кэшу ответов. Значения «Не передаётся» не включаются в расчёт как нули.

При отсутствии стоимости от шлюза в аналитике отображается «не получено». Это отсутствие сведений, а не нулевая цена.

При превышении или исчерпании лимита HTTP 429 содержит описание превышенного ограничения. Для разбора сохраните фактический ответ, модель, область и время. Не определяйте остаток квоты или срок снятия ограничения по одному коду ответа.

Учёт без двойного списания

Продуктовый учёт построен по схеме «резерв квоты → вызов шлюза → фиксация фактического потребления». Если резерв не получен, вызов не выполняется. События проходят через ValKey, а фоновый обработчик записывает результат в продуктовый журнал PostgreSQL.

Резерв, событие и запись журнала представляют разные части одной операции. Их нельзя суммировать как независимое потребление. После сбоя повторная обработка не должна приводить к двойному списанию. Отсутствие записи в аналитике само по себе не доказывает, что запрос не выполнялся: путь записи включает фоновый обработчик.

Очистка ValKey и ручная корректировка продуктового журнала не используются как способ исправления расхождений. При незавершённом резерве или неполной записи сохраняют сведения об исходном запросе и передают инцидент по главе 7.

Расписание и готовность модели

Продуктовое расписание задаёт окна доступа к запросам и проверяется до обращения к модели. Разрешённое окно не гарантирует готовности экземпляра, а готовая модель не отменяет запрет доступа по времени. Расписание доступа не управляет освобождением GPU.

Создание расписания доступа к моделям

1. Откройте «Организации» → нужная организация → «Расписания доступа».
2. В блоке «Расписания доступа к моделям» проверьте текущий список. Сообщение «Расписаний пока нет» означает пустой список этой вкладки, а не отсутствие любых ограничений доступа.

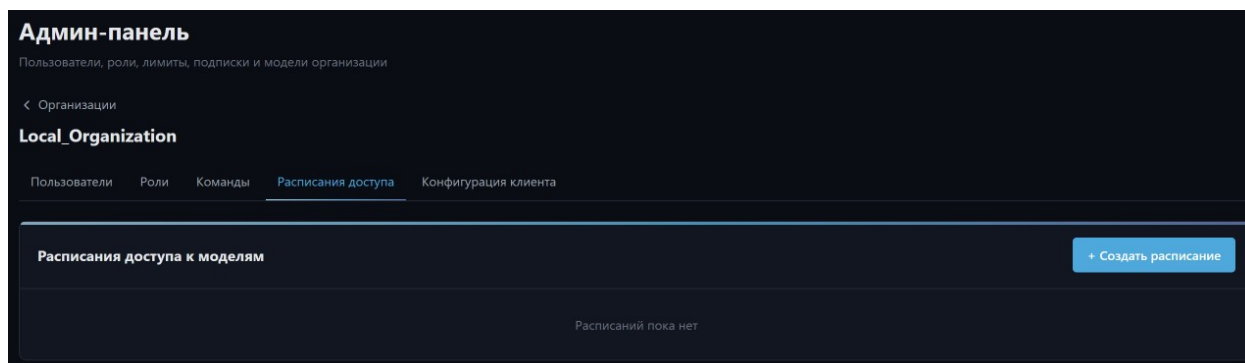


Рисунок 19 — Расписания доступа: пустой список

3. Нажмите «+ Создать расписание». В окне «Новое расписание доступа» проверьте «Область действия» и «Модель». Значения «Вся организация» и «Все модели» задают общую область; используйте её только для общего расписания.
4. Если требуется отдельная модель, сопоставьте доступный выбор с пояснением под полем: список берётся из лимитов текущего тарифа организации. Если список пуст, проверьте подписку; не заменяйте ограничение одной модели значением «Все модели» без согласования области.
5. Заполните обязательный «Часовой пояс *» по утверждённому расписанию. Проверьте флажок «Расписание включено». Начальные значения формы не являются обязательным часовым поясом или режимом поставки.
6. В блоке «Еженедельные интервалы» выберите нужные дни Пн–Вс и задайте время начала и окончания. Для дополнительных строк используйте «+ Добавить интервал», для ненужной строки — «Убрать». Сверьте каждый день и интервал в выбранном часовом поясе.
7. Укажите «Действует с» и «Действует по» в соответствии с периодом действия. Блок «Исключения» содержит кнопку «+ Добавить исключение».
8. Сверьте область, модель, часовой пояс, включённость, интервалы и даты. Нажмите «Создать» либо «Отмена» для выхода без отправки.
9. Повторно откройте «Расписания доступа» и проверьте наличие записи. Проверку фактического разрешения внутри окна и отказа вне окна выполняйте отдельно по согласованному сценарию, а не по одному факту сохранения формы.

Расписание доступа не является командой развёртывания или гарантией готовности модели. Конфигурация клиента и управление моделями рассматриваются в главе 4.

Новое расписание доступа

✕

Область действия

Вся организация

Модель

Все модели

Часовой пояс *

Europe/Moscow

Список моделей берётся из лимитов текущего тарифа организации - в нём пока не задано ни одной модели

☒ Расписание включено

Еженедельные интервалы

Пн

Вт

Ср

Чт

Пт

Сб

Вс

09:00

—

18:00

Убрать

+ Добавить интервал

Действует с

📅 Выберите дату

Действует по

📅 Выберите дату

Исключения

+ Добавить исключение

Отмена

Создать

1. Выберите «Аналитика» в боковом меню. На странице «Аналитика потребления» задайте период кнопкой 7, 30 или 60 дней либо датами «С» и «ПО».
2. Ограничьте выборку фильтрами «КОМАНДА», «МОДЕЛЬ», «КАЧЕСТВО». Для сравнения отчётов сохраняйте одинаковые значения фильтров; названия моделей в истории не подтверждают их текущую доступность.
3. Просмотрите показатели «Расходы», «Токенов всего», «Запросов», Error rate, «Точный учёт» и Response cache hits. Не объединяйте качество учёта, кэш ответов и категории кэша токенов в один показатель.
4. Значение «Не определено» у качества учёта сохраняйте в результатах анализа; оно не означает точный учёт или нулевое потребление.

5. В «Динамика по дням» поочерёдно выберите «Расходы», «Токены» или «Запросы» для нужного сравнения.
6. Во вкладке «По моделям» сопоставьте модель, расходы, токены, число запросов и ошибок. Для других разрезов используйте «По пользователям» или «Запросы».
7. Для разбора отказов откройте «Ошибки» и выполните процедуру раздела 7.1. Число в названии вкладки относится к показанной выборке и не доказывает полный итог ошибок системы.
8. Для загрузки нажмите «Скачать CSV». Проверьте полученный файл: период, состав данных и соответствие фильтрам. Храните его в разрешённом контуре; перед передачей исключите лишние сведения об участниках.

Вкладка ошибок поддерживает поиск. Нажатие на строку не открывает подробную карточку ошибки.

При сопоставлении отчётов важны одинаковые период, организация и модель, а также полнота поступления событий. Неизвестную стоимость необходимо сохранять как отсутствие сведений, не заменяя нулём ради получения итога.

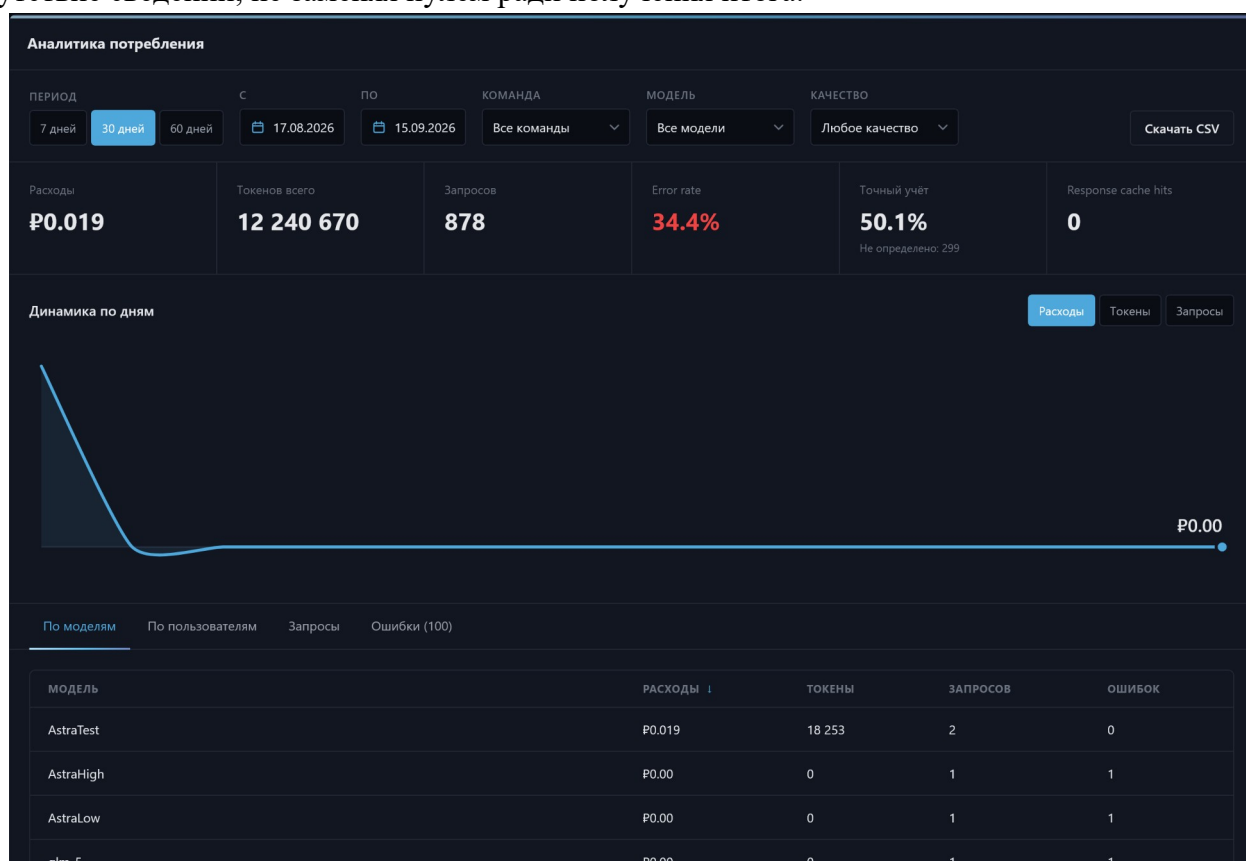


Рисунок 21 — Аналитика потребления: фильтры, показатели и динамика

Аудит и источники технического состояния

Журнал аудита фиксирует, кто выполнил действие, что произошло, когда и с каким результатом. В него не включаются ключи, тела запросов и тексты обращений к модели. Сквозной идентификатор запроса связывает аудит, учёт и разбор инцидента.

Необходимо различать следующие источники:

- продуктовый журнал — фактическое потребление и основа аналитики;
- аудит — действия и события безопасности;

- статусы, журналы и события развёртываний — состояние моделей и операций платформы;
- метрики оператора, шлюза и вычислительных ресурсов — нагрузка и техническое состояние.

Успешная запись в одном источнике не заменяет сведений другого. Например, доступность процесса не подтверждает полноту продуктового учёта, а отсутствие ошибок модели не подтверждает корректность организационных прав.

Живость и готовность

Серверная часть может работать только в режиме управления, только в режиме исполнения запросов или в совмещённом режиме. Проверки имеют разный смысл:

Проверка	Что означает
/health/live	Процесс запущен; результат не зависит от внешних сервисов
/health/ready	Доступны зависимости, необходимые выбранному режиму работы

Готовность учитывает базу данных и ValKey для учёта и кэша, шлюз моделей для исполнения запросов, HTTP-адаптер развёртываний для управления. Если необходимая зависимость недоступна, экземпляр не считается готовым к нагрузке. Успешная проверка живости не опровергает такой отказ готовности. Адрес публикации проверок и доступ к ним определяются поставкой; приведённые пути не являются указанием открыть служебные маршруты внешним пользователям.

Мониторинг и обслуживание журналов

Платформенные средства наблюдаемости включаются отдельно. Метрики оператора и шлюза предоставляются на отдельных точках. Работа без VictoriaMetrics или KEDA при выключенной соответствующей функции не означает наличие заменяющих панелей или автоматического масштабирования.

Периодичность сбора, доступ, сроки хранения журналов и пороги оповещений задаются эксплуатационным регламентом. Хранение диалогов отделяется от аудита: содержимое сохраняется только по разрешённой политике, а аудит ведётся без текстов запросов и секретов.

6.7. Управление конфигурацией и данными

Конфигурация клиента

Для централизованных настроек клиента откройте организацию → «Конфигурация клиента». Подробный порядок работы с областями и полями приведён в главе 4. Выбор модели в этой форме не развёртывает её и не заменяет проверку доступа.

Границы конфигурации

Продуктовая конфигурация определяет доступ, подписки и политики; платформенная — работу ресурсов Kubernetes, среды исполнения и включённых подсистем. Консоль использует продуктовый API, а серверная часть взаимодействует с ресурсами через HTTP-адаптер. Ручная правка кластерного ресурса не заменяет изменение продуктовой настройки.

Сохраняйте согласованный набор версий компонентов и применённую конфигурацию без значений секретов. Учитывайте включённые подсистемы и внешних поставщиков.

PostgreSQL используется совместно, но с разными схемами и ролями: продуктовый журнал относится к Астра ИИ, платформенный учёт и диалоги — к контуру исполнения моделей. ValKey содержит не только кэш ключей, но и резервы квот, события учёта и контекст запросов. Поэтому все данные ValKey нельзя заранее признать безопасно удаляемым кэшем.

Сохранность и восстановление данных

Резервное копирование и восстановление выполняются по эксплуатационному регламенту. Копии и секреты хранятся в защищённом контуре с ограничением доступа. Восстановление отдельных компонентов не означает согласованного восстановления организаций, ключей, подписок, резервов и продуктового учёта.

Запуск процессов после восстановления проверяется отдельно от доступа к моделям и корректности учёта. Исходный запрос не должен учитываться повторно. Очистка ValKey или ручная правка PostgreSQL не заменяют восстановление согласованного состояния.

6.8. Ограничения обновления и отката

При обновлении сохраняется имя узла входа. Обновление реплик модели, при котором новая реплика достигает готовности до удаления старой, не заменяет обновление всей поставки и миграцию данных.

Наличие предыдущей версии пакета не доказывает безопасность отката базы данных. Возврат версии выполняется только по регламенту, учитывающему совместимость данных и конфигурации. Общая команда Helm не заменяет такой регламент.

После изменения проверяются готовность нужных режимов, доступ к консоли и моделям, действие политик, продуктовый учёт и сохранение имени узла входа.

6.9. Ограничения вывода из эксплуатации

Вывод из эксплуатации затрагивает доступ, ключи, вычислительные ресурсы и данные. Отсутствие подписки закрывает запросы и каталог моделей, но сохраняет управление организацией. Отзыв одного ключа не прекращает все каналы доступа.

Удаление данных выполняется как отдельная разрешённая операция с учётом состава и сроков хранения. Удаление развёртывания или пакета не используется как универсальная процедура удаления постоянных томов, объектных хранилищ, журналов, диалогов и резервных копий. Сохранность данных контролируется отдельно от состояния компонентов.

При прекращении обслуживания фиксируются затронутые клиенты и интеграции, состояние доступа и результаты разрешённых операций с данными. Секреты в отчёт не включаются.

7. Типовые проблемы и их решения

Глава описывает первичную диагностику серверного ХАБ: локализацию отказа, сбор безопасных сведений и передачу инцидента ответственному за затронутую функцию. Восстановительные изменения выполняются отдельно, по эксплуатационному регламенту.

Нажатие строки ошибки не раскрывает подробности; разбор начинается с таблицы. Необязательные подсистемы рассматриваются только при их включении.

7.1. Порядок первичной диагностики

Первичная диагностика устанавливает участок отказа и собирает фактические сведения для дальнейшего разбора. Отказ запроса не доказывает неисправность модели: запрос может быть отклонён до её вызова.

Последовательность безопасных проверок

1. **Зафиксируйте наблюдение до изменения состояния.** Запишите время и часовой пояс, выполнявшуюся операцию, ожидаемый и фактический результат, полученный HTTP-код и обезличенное сообщение, если они доступны. Сохраните идентификатор запроса или операции без значений ключей. Если идентификатор не виден, отметьте это; не создавайте собственный идентификатор под видом системного.
2. **Определите влияние.** Затронуты вход в консоль, управление, выполнение запросов, учёт? Проблема касается одного пользователя, организации, модели либо нескольких? Используйте уже имеющиеся наблюдения и разрешённые сведения, не обходите права для проверки чужих организаций.
3. **Зафиксируйте конфигурацию.** Запишите версию выпуска и компонентов, режим серверной части — управление, исполнение или совмещённый; тип модели — локальная или внешняя; включённые опции. Используйте эксплуатационную конфигурацию.
4. **Сопоставьте время отказа с изменениями.** Отметьте смену ключа, роли, членства, подписки, политики, расписания, конфигурации модели или выпуска. Совпадение во времени — основание для проверки, но не доказанная причина.
5. **Разделите живость, готовность и результат операции.** По проверкам ниже определите, работает ли процесс и доступны ли зависимости его режима. Затем переходите к соответствующему разбору в разделах 7.2–7.4.
6. **Найдите последнюю подтверждённую границу.** Например: вход принят, продуктовая проверка завершена, запрос передан шлюзу, модель начала ответ, учёт зафиксирован. Отсутствие диагностического события не доказывает, что этап не выполнялся.
7. **Соберите пакет и назначьте дальнейший разбор** по разделу 7.5. Прекратите самостоятельные проверки, если для продолжения нужны дополнительные полномочия, секреты или изменение состояния без регламента.

Поиск ошибки в административной панели

1. В боковом меню выберите «Аналитика». Установите период 7, 30 или 60 дней либо даты «С» и «ПО», охватывающие отказ. Задайте «КОМАНДА», «МОДЕЛЬ» и, если нужно, «КАЧЕСТВО»; не сужайте выборку неизвестным значением качества.
2. Откройте вкладку «Ошибки». Для поиска используйте пользователя или статус.
3. Сопоставьте «Найдено» и «Всего» для текущей выборки. Для просмотра страниц используйте «Назад» и «Вперёд», ориентируясь на указатель страницы. Недоступная «Назад» на первой странице — обычное состояние навигации.
4. Найдите строку по сочетанию ДАТА, МОДЕЛЬ и ПОЛЬЗОВАТЕЛЬ. Запишите время вместе со смещением часового пояса, СТАТУС и ОШИБКА. Сопоставьте с временем на стороне клиента с учётом часовых поясов, не сравнивая только часы и минуты.

- 5. Если показаны failed и Astra AI Hub inference request failed, зафиксируйте их буквально. Это общий результат, а не диагноз отказа модели, шлюза, квоты или подписки. Значение failed в таблице не является HTTP-кодом.
- 6. Нажатие строки не открывает подробную карточку. В таблице отсутствует идентификатор запроса. Для сопоставления с журналами передайте время, модель и допустимое обозначение пользователя; полученный из журнала идентификатор добавьте с указанием источника.
- 7. При необходимости нажмите «Скачать CSV» и проверьте полученный файл по разделу 6.6. Для обращения оставьте только нужный фрагмент и исключите секреты и лишние данные участников.
- 8. Продолжите проверку доступа по разделу 7.2, состояния модели — по разделу 7.3 или учёта — по разделу 7.4. Не повторяйте запрос к модели только для получения более подробной ошибки.

Число в названии вкладки, счётчики и пагинация описывают доступную выборку, а не гарантированный итог всех ошибок за всё время. Пустая выборка требует проверки периода, фильтров и прав, а не вывода об отсутствии отказов.

По моделямПо пользователямЗапросыОшибки (100)

Поиск по пользователю, статусу.Найдено: 100

ДАТА	МОДЕЛЬ	ПОЛЬЗОВАТЕЛЬ	СТАТУС	ОШИБКА
2026-09-15T12:44:00.130635+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:56.951025+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:55.177558+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:54.167244+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:53.640504+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:53.268500+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:11.212658+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:07.910386+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:06.241292+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:05.129494+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:04.597855+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T12:43:04.192195+00:00	test/deepseek-v4-flash		failed	Astra AI Hub inference request failed
2026-09-15T11:42:35.790379+00:00	gpt-5.4-mini		failed	Astra AI Hub inference request failed
2026-09-15T11:39:52.458464+00:00	gpt-5.4-mini		failed	Astra AI Hub inference request failed
2026-09-15T11:39:52.412720+00:00	gpt-5.4-mini		failed	Astra AI Hub inference request failed

Всего: 100Назад1 / 7Вперед

Рисунок 22 — Аналитика: поиск и таблица ошибок

Живость и готовность серверной части

Серверная часть предоставляет две проверки:

Проверка	Значение	Диагностический вывод и ограничение
/health/live	Процесс запущен; проверка не зависит от	Успешный результат не подтверждает готовность

Проверка	Значение	Диагностический вывод и ограничение
	внешних сервисов	обслужить запрос и исправность модели
/health/ready	Доступны зависимости, требуемые выбранным режимом	Неготовность требует разбора зависимостей; это не автоматическое основание перезапустить процесс

Готовность учитывает базу данных и ValKey для учёта и кэша; в режиме исполнения — шлюз моделей, в режиме управления — HTTP-адаптер. Проверки выполняются из разрешённой служебной точки. Их не публикуют наружу.

Наблюдение	Проверка	Куда продолжить
До точки входа нельзя получить ответ	Проверить утверждённый маршрут, доступность периметра и наличие наблюдений на APISIX; не менять маршрут на прямое обращение к шлюзу	Ответственные за входной контур; при неясной границе — совместный разбор
Живость подтверждена, готовность — нет	Установить режим экземпляра и недоступную зависимость по разрешённой диагностике	Раздел 7.4
Готовность подтверждена, отдельный запрос отклонён	Проверить доступ, лимиты, политику, модель и конкретную операцию	Разделы 7.2–7.3
Живость не подтверждается	Отличить недоступность пути проверки от остановки процесса по сведениям эксплуатации	Ответственный за размещение компонента; без перезапуска для диагностики

7.2. Вход и отказ доступа к модели

Путь запроса разделяет аутентификацию на APISIX и авторизацию в серверной части. APISIX проверяет пользователя или сервисный ключ и формирует доверенные сведения о личности; роли организации он не проверяет. Успешный вход не означает разрешение любой операции управления или вызова модели.

Диагностика сопоставляет симптом с состоянием доступа и компонентов. HTTP-код и общее сообщение об ошибке не заменяют установление причины.

Симптом	Возможная область причины	Что проверить без изменения политики	Дальнейшее действие
Не удаётся войти в	Интеграция OIDC,	Где прекращается	Передать

Симптом	Возможная область причины	Что проверить без изменения политики	Дальнейшее действие
консоль	поставщик идентификации, APISIX или путь до него	вход; виден ли результат аутентификации; затронута одна учётная запись или несколько; время и идентификатор события	продуктовой команде сведения о границе отказа; не прикладывать токен и не подставлять заголовки личности вручную
Вход выполнен, операция управления запрещена	Недостаточные полномочия либо неверная область операции	Разрешения роли и организацию; относится ли операция к платформенному администрированию	Проверить назначение прав по главам 5–6; не выдавать расширенную роль только для обхода отказа
Управление организацией доступно, каталог моделей и запросы — нет	Подписка неактивна	Состояние подписки и область организации	Разобрать подписку в продуктивном контуре: её отсутствие не снимает права управления организацией
Персональный запрос отклонён	Членство, подписка, разрешённая модель, расписание, квота	Каждое основание отдельно, с учётом организации, команды и пользователя	Передать конкретное несоответствие владельцу политики; не объявлять модель неисправной без подтверждения её вызова
Сервисный запрос отклонён	Ключ либо политика организации	Тип и состояние ключа, его организацию, разрешённые модели, подписку, расписание и ограничения организации	Не применять к сервисному ключу командные ограничения: на него действуют только политики организации
Один тип запроса проходит, другой отклоняется	Неподдерживаемый маршрут или формат	Операцию, разрешённые маршруты и	Передать несоответствие ответственному за

Симптом	Возможная область причины	Что проверить без изменения политики	Дальнейшее действие
		возможности выбранной модели	API; не обходить продуктовую точку входа
Доступ зависит от времени	Окна продуктового доступа	Время с часовым поясом и применимые расписания	Сопоставить отказ с окном доступа; готовность модели проверять отдельно

Права администратора организации не дают права платформенного администратора. Для отказа управления развёртыванием следует сначала проверить именно эту границу, а не увеличивать ресурсы модели.

Проверка доступных настроек без их изменения

1. Откройте нужную организацию → «Пользователи» и сверьте статус, роли и команду. Во вкладке «Роли» проверьте фактические разрешения; название Viewer не означает универсальный просмотр.
2. В «Команды» откройте «Состав» нужной команды и проверьте участника. Не используйте «Добавить», «Убрать» или «Деактивировать» как диагностическую пробу.
3. В «Подписки» сверьте текущий тариф, начало и срок действия, условия пользователей, ключей и моделей. «Оформить» меняет условия использования и не является проверкой доступности.
4. Во вкладке организации «Расписания доступа» сопоставьте доступные записи с моментом отказа. Для новых настроек порядок полей приведён в разделе 6.5; не создавайте расписание ради диагностики и не предполагайте правила пересечения окон.
5. Для машинного запроса откройте «Сервисные API-ключи», проверьте запись по названию, дате и статусу. Не запрашивайте значение ключа и не отзывайте его для проверки гипотезы.
6. В «Квоты» посмотрите результат выбранного публичного имени и соответствующего режима Streaming по следующему подразделу. Если проблема относится к выбору модели в конфигурации клиента, используйте главу 4: список этой формы не является перечнем готовых развёртываний.

Ключ недействителен после изменения

смена пользовательского ключа делает старый недействительным; отзыв отключает клиента; после изменения обновляется кэш определителя ключей. Разрешение ключей на APISIX использует ValKey с запасным чтением PostgreSQL. Секрет не должен попадать в журналы и ответы кластерного интерфейса.

Проверка:

1. Зафиксировать тип ключа, доступный несекретный идентификатор, владельца, организацию, время и результат операции смены или отзыва.

2. Сопоставить факт обновления настройки клиента с операцией смены, не запрашивая старое или новое значение ключа. Отклонение старого значения после смены является штатным результатом.
3. Сопоставить состояние ключа в продукте, результат операции и состояние клиента платформы. Различать завершённую смену и частично выполненную операцию.
4. При расхождении передать сведения ответственному за продуктовый учёт и определение ключей. Запасное чтение базы не заменяет остальные функции ValKey.
5. При противоречии передать инцидент в продуктовый контур, а при расхождении с клиентом платформы — на совместный разбор.

Не выпускать новые ключи серий «до успеха», не очищать кэш и не редактировать записи базы вручную. Если завершённость изменения неизвестна, действовать по разделу 7.4.

Превышен лимит

При превышении или исчерпании лимита возвращается **HTTP 429 с описанием превышенного ограничения**. Для разбора используйте фактически указанное ограничение и его область.

Для разбора зафиксируйте фактически указанное ограничение, его область, модель, период, время начала окна, если оно доступно, и наблюдаемое потребление. Разделяйте остаток квоты, частоту обращений и число одновременных запросов.

1. Откройте «Квоты», введите публичное имя затронутой модели, установите Streaming в соответствии с отказавшим запросом и нажмите «Показать».
2. Сохраните текст результата и время последнего обновления. Сообщение «Для этой модели тариф не задаёт ограничений использования.» относится только к тарифу выбранной модели и режима. Оно не опровергает HTTP 429 от другого уровня, не отменяет расписание и не подтверждает готовность модели.
3. При сравнении обычного и потокового ответа заново нажмите «Показать» после изменения Streaming. Совпадение результата одной модели не доказывает совпадения всех режимов и моделей.
4. В «Контракт учёта токенов» зафиксируйте ревизию, статус и значения категорий. «Не передаётся» не заменяйте нулём; «Не проверен» не выдавайте за подтверждение точности.
5. По «К подписке» сопоставьте результат с текущим тарифом. Если ответ не содержит конкретного превышения, передайте текст фактического ответа и результаты просмотра продуктовой команде; не меняйте тариф на основании общего сообщения об ошибке.

При штатном ограничении согласуйте с владельцем нагрузки снижение интенсивности или использование в разрешённом окне. Изменение тарифа и политики выполняется как отдельная разрешённая операция. Не обходите ограничение сменой ключей, организаций или маршрута.

Сработала защита содержимого

защита содержимого — включаемая опция шлюза. При её включении политики могут блокировать, маскировать или только регистрировать срабатывание. Обязательную политику клиент не может снять параметрами запроса. Блокировка нормализуется как отказ политики

безопасности, а не ошибка модели; события аудита и метрики не содержат тел запросов и ответов.

Рекомендуется проверить, включена ли защита в поставке, какой итоговый набор политик применён к модели и клиенту, фазу проверки и фактически выполненное действие — если эти сведения доступны. Отдельно установить, произошло срабатывание правила или отказ детектора: отказ детектора по умолчанию приводит к запрету, поэтому блокировка сама по себе не доказывает наличие запрещённого содержимого.

Передайте сведения о правиле и событии владельцу политики безопасности. Для технического отказа детектора привлечите ответственного за его эксплуатацию и шлюз. Не отключайте обязательную защиту и не переводите её в разрешающий режим как стандартное исправление. Проверку спорного срабатывания согласуйте отдельно на разрешённых обезличенных материалах, а не на опубликованном пользовательском запросе.

7.3. Модель недоступна или развёртывание не готово

Управление проходит через продуктовый API и HTTP-адаптер; оператор Kubernetes согласует фактическое состояние с заданным. Создание записи или принятие операции не означает готовность модели обслуживать запросы. Для диагностики используются состояние, журналы, события, метрики, перечень GPU и оценка размещения через продуктовый API.

Симптом	Проверяемая гипотеза	Безопасная проверка	Ответственный за дальнейший разбор
Создание отклонено до появления ресурса	Проверка конфигурации либо полномочий	Результат проверки спецификации, права платформенного администратора, последнюю принятую конфигурацию; не переносить секреты из неё	Астра ИИ; при ошибке преобразования в ресурс — совместно с владельцем адаптера
Операция принята, модель не готова	Согласование ещё идёт или нарушено	Сопоставить результат продуктовой операции, ресурс и его наблюдаемое состояние, события оператора за один интервал	Ответственные за HTTP-адаптер и исполнение моделей совместно, если неясно, где остановилась передача
Подготовка весов не завершается	Источник, доступ к нему, загрузчик, том или кэш	Состояние загрузки и событий хранения; наличие разрешённой ссылки на секрет,	Ответственный за эксплуатацию платформенного контура; доступ к внешнему

Симптом	Проверяемая гипотеза	Безопасная проверка	Ответственный за дальнейший разбор
		но не его содержимое	хранилищу — с его ответственным
Модель не размещается либо обновление не завершается	Ограничения GPU или другие требования размещения	Перечень доступных GPU, предварительную оценку размещения, состояние новой реплики и события; оценка не гарантирует успешное размещение	Ответственный за эксплуатацию платформенного контура
Модель готова, но запросы медленные	Очередь, нагрузка, среда исполнения либо внешняя зависимость	Доступные задержки первого и последующих токенов, общую задержку, очередь, нагрузку GPU; сравнить один период и модель	Ответственный за эксплуатацию платформенного контура; при задержке до шлюза — совместно с Астра ИИ
Недоступна только внешняя модель	Конфигурация внешнего поставщика, путь до него или его сервис	Тип модели, состояние поставщика, обезличенный результат обращения и время; наличие настроенной ссылки на секрет	Астра ИИ по регистрации модели; Ответственный за эксплуатацию платформенного контура по исполнению, далее ответственный за взаимодействие с поставщиком

Для внешней модели отсутствие локальных подов исполнения нормально: запрос проксируется внешнему OpenAI-совместимому источнику. Для локальной модели загрузка весов и согласование оператора являются отдельными этапами. Не удаляйте веса, тома или ресурсы модели для проверки гипотезы.

При обновлении новая реплика достигает готовности до удаления старой. Не заменяйте этот порядок ручным удалением работающей реплики ради освобождения GPU. Разбор нехватки ресурсов передайте платформенному контуру.

Холодный старт

при запуске модели из нулевого масштаба шлюз удерживает запрос в пределах отведённого времени ожидания либо возвращает явный **HTTP 503**. Переходом $0 \leftrightarrow 1$ управляет оператор, масштабом $1 \leftrightarrow N$ — KEDA при включении этой опции.

Сопоставьте время запроса с переходом из нулевого масштаба, событиями загрузки и достижением готовности. Зафиксируйте фактическое время ожидания и результат. HTTP 503 сам по себе не доказывает холодный старт.

Время ожидания и повторные запросы ограничиваются настройками используемого клиента и сервера. Не увеличивайте время произвольно и не запускайте параллельные запросы «для прогрева»: они увеличивают нагрузку и потребление.

Неподдерживаемая операция

Если снятие модели с публикации не поддерживается, операция возвращает явный отказ. Зафиксируйте операцию, выпуск, сообщение без секретов и фактическое состояние. Запрошенное действие не означает выполненное снятие с публикации.

Если получен успех, а состояние не изменилось, сначала сопоставьте ответ с состоянием операции: принятие и завершение — разные результаты. Затем сопоставьте состояние продукта и платформы. Не заменяйте снятие с публикации удалением ресурса: это другая операция с иными последствиями.

Потоковый ответ и WebSocket

Поток ответа модели (SSE) и WebSocket-консоль развёртывания — разные соединения. Первый связан с выполнением запроса и учётом, второй — с интерактивным доступом и передачей состояния. Токен WebSocket проходит OIDC на APISIX и не передаётся следующему сервису как реквизит доступа.

Для разрыва соединения рекомендуется собрать:

- тип соединения, операцию, модель или развёртывание, время начала и разрыва;
- был ли принят запрос и началась ли передача данных;
- последнюю подтверждённую границу: клиент — APISIX — продуктовая серверная часть — адаптер либо шлюз — исполнитель;
- фактический результат закрытия и доступный код, без назначения ему причины по догадке;
- для SSE — получен ли финальный фрагмент со сведениями об использовании токенов и имеется ли запись учёта;
- для WebSocket — прошла ли аутентификация и на каком участке подтверждено соединение, без копирования токена и полного заголовка подпротокола.

Не прикладывайте необработанный сетевой архив браузера. Он может содержать токены, ключи и пользовательское содержимое. Разрыв у клиента не доказывает остановку исполнения; отсутствие финального фрагмента не означает нулевое потребление. Перед повтором запроса с неопределённым исходом проверьте состояние учёта и допустимость повтора.

7.4. Сбой зависимостей и рассогласование учёта

Управление и исполнение могут работать отдельно. Поэтому недоступность консоли, невозможность изменения модели и отказ запросов к модели следует фиксировать отдельно. Проверка готовности оценивает зависимости конкретного режима, а не все компоненты системы одновременно.

Зависимость или участок	Возможное проявление, не готовый диагноз	Сопоставление	Ограничение безопасной диагностики
PostgreSQL	Неготовность серверной части, недоступные либо неполные данные аналитики	Режим процесса, доступность нужной базы, продуктовый журнал за интервал, состояние записи учёта	Разделять продуктовый журнал и опциональный учёт платформы; не исправлять строки базы вручную
ValKey	Неготовность, проблемы разрешения ключей, резерва или переноса событий учёта	Состояние зависимости, этап «резерв — вызов — фиксация», состояние фонового обработчика	Не очищать кэш, резервы или события; запасное чтение ключа из базы не заменяет остальные функции ValKey
HTTP-адаптер	Неготовность управления, отсутствие результата операции с развёртыванием	Продуктовую операцию, результат адаптера и наличие соответствующего состояния платформы	При неизвестном результате не выполнять создание заново
Шлюз моделей	Неготовность исполнения или отказ передачи запроса	Дошёл ли разрешённый запрос до шлюза; затронуты все модели или отдельная; состояние исполнителя	Готовность управления не подтверждает готовность исполнения; не направлять клиента в обход продуктовых проверок
Средства наблюдаемости	Нет панелей или метрик при работающих запросах	Включены ли соответствующие опции, доступен ли сбор данных и какой период виден	Отсутствие необязательного набора средств наблюдения не считать отказом модели; отсутствие данных не равно отсутствию ошибки

Учёт после ответа модели

продукт резервирует квоту, вызывает шлюз и фиксирует фактическое потребление. События проходят через ValKey, фоновый обработчик записывает их в PostgreSQL. Отказ

резерва означает отсутствие вызова. Учёт не должен дублироваться. Если стоимость от шлюза не получена, в аналитике показывается «не получено» — это не нулевая цена.

Если пользователь получил ответ, но запись отсутствует, неполна или выглядит повторной, рекомендуется:

1. Проверить область видимости аналитики, фильтры организации, пользователя, модели и периода. Сравнение разных областей не доказывает потерю данных.
2. Найти исходный случай по времени с часовым поясом, модели и пользователю. Если в разрешённых журналах доступен сквозной идентификатор, сопоставить по нему резерв, факт вызова, сведения об использовании из ответа, событие и запись продуктового журнала. В таблице ошибок отсутствует идентификатор запроса. Для каждого этапа указать результат и доступность сведений.
3. Различать продуктовый журнал и платформенный учёт, если последний включён. Их разные схемы и роли не означают, что одинаковые записи нужно суммировать.
4. Проверить признаки работы фоновой обработчика и зависимостей за нужный интервал. Задержка интерфейса сама по себе не является доказательством потери или успешной записи учёта.
5. Передать расхождение продуктовой команде, при необходимости совместно с владельцем шлюза. Сохранить доступный идентификатор исходного запроса либо набор времени, модели и пользователя для поиска; не инициировать повтор как способ «дописать учёт».

Неполные сведения в аналитике и контракте учёта

1. В «Аналитика» сверить период и фильтры «КОМАНДА», «МОДЕЛЬ», «КАЧЕСТВО»; сопоставить итог и «По моделям» в одной выборке.
2. В «Квоты» выбрать то же публичное имя и нужный Streaming, нажать «Показать». Сохранить ревизию, статус и время обновления контракта.
3. Отдельно записать качество, источник и допустимый контроль для «Вход без кэша», «Запись prompt cache», «Чтение prompt cache», «Выход». Сочетание «Не передаётся» / прочерк / «Нет» не доказывает ни нулевое потребление, ни отсутствие всех видов учёта.
4. Разделять «Не определено» у качества аналитики, «Не проверен» у контракта и отсутствие стоимости. Не заменять пропуски нулями и не объявлять общий итог точным, если необходимые категории не передаются.
5. Передать различия вместе с областью выборки и исходным случаем. Проверку фоновой учёта продолжать по предыдущему подразделу, без очистки ValKey и ручной корректировки PostgreSQL.

Частично выполненная операция

Истечение ожидания или разрыв соединения не устанавливают, был ли объект создан.

Зафиксируйте идентификатор операции, тип объекта, его несекретный идентификатор, время и последнюю известную стадию. Сопоставьте продуктовую запись, результат адаптера и состояние платформы доступными средствами просмотра.

Не повторяйте создание и не удаляйте объект без проверки исходной операции. Передайте инцидент по разделу 7.5. Идемпотентность оператора не гарантирует безопасность любого повторного запроса администратора.

7.5. Сбор и передача диагностики

Для диагностики используются аудит безопасности, состояния и события развёртываний, журналы компонентов, учёт и метрики. Аудит содержит сведения «кто, что, когда, результат», но не ключи и тела запросов. Сквозной идентификатор связывает сведения одного запроса.

Состав диагностического пакета

Сведения	Что включить
Контекст поставки	Версия, режим серверной части, включённые опции, локальная или внешняя модель; неизвестные значения явно отметить
Влияние	Затронутая функция, область организаций и моделей, начало и длительность наблюдения, сохраняющиеся работоспособные функции
Симптом	Операция, ожидаемый и фактический результат, фактический HTTP-код и обезличенное сообщение
Корреляция	Доступные идентификаторы запроса, события или операции с указанием источника; при их отсутствии — время с часовым поясом, модель и допустимое обозначение пользователя из таблицы; несекретные идентификаторы объектов
Состояние	Результаты живости и готовности с указанием экземпляра/роли в разрешённом объёме; состояние модели и зависимостей
Подтверждения	Минимальные выдержки из аудита, событий, журналов и метрик за нужный интервал, с указанием источника и времени
Выполненные действия	Проведённые проверки, фактические результаты и разрешённые изменения
Задача дальнейшего разбора	Последняя подтверждённая граница, проверяемая причина и требуемое решение

Если идентификаторы организации, пользователя или ресурса нельзя передавать получателю, согласуйте устойчивые заменители. Они должны одинаково обозначать объект во всех приложениях; таблица соответствий остаётся в разрешённом контуре. Способ обезличивания и возможность передачи исходных идентификаторов требуют согласования с требованиями безопасности организации.

Запрещённые вложения

Не включайте в обычное обращение:

- значения пользовательских, сервисных и внутренних ключей, токены сеансов и доступа, пароли, файлы секретов и другие реквизиты подключения;

- полные заголовки авторизации, содержимое файлов cookie и необработанные сетевые загрузки;
- тела запросов, пользовательские инструкции, ответы моделей, диалоги и иное пользовательское содержимое без отдельного разрешённого порядка;
- полные конфигурации, дампы баз и хранилищ «на всякий случай»;
- посторонние адреса и персональные контакты.

Перед передачей просмотрите текст, снимки экрана и вложения. Маскируйте не только известные названия полей: сообщение внешнего поставщика или снимок интерфейса также может раскрывать секрет. Сохраняйте причинно значимые время, код и связь событий, но не ценой раскрытия запрещённых данных. При обнаружении утечки остановите распространение пакета и действуйте по согласованному порядку информационной безопасности; не копируйте секрет в описание инцидента.

Маршрутизация обращения

Обращение направляется ответственному за затронутую функцию по эксплуатационному регламенту. Границы контуров:

Область	Кому адресовать первичный разбор	Когда нужен совместный разбор
Консоль, вход, права, членство, подписки, ключи, квоты, продуктовый учёт	Астра ИИ	При расхождении с состоянием клиента платформы, учётом или обработкой на шлюзе
Оператор, загрузка и кэш весов, GPU, размещение, среда исполнения, платформенная наблюдаемость	Ответственный за эксплуатацию платформенного контура	Если продуктивное состояние расходится с платформенным или запрос не доходит до исполнителя
Общий шлюз	Астра ИИ по политикам, учёту, лимитам и расписаниям; Ответственный за эксплуатацию платформенного контура по инфраструктуре и поставке	Если граница не доказана, сразу обе стороны с единым идентификатором запроса
PostgreSQL и ValKey	Ответственный за затронутую функцию и эксплуатацию зависимости	При нарушении продуктового учёта и связей с платформой
HTTP-адаптер	Ответственный за интеграцию продуктового API с Kubernetes	При расхождении результата операции и состояния ресурса
Защита содержимого	Ответственный за политику безопасности	При отказе детектора или применения политики — совместно с эксплуатацией шлюза
Внешняя модель	Владелец регистрации	Взаимодействие с

Область	Кому адресовать первичный разбор	Когда нужен совместный разбор
	модели и эксплуатации шлюза	поставщиком — через назначенного ответственного, без передачи ему секретов ХАБ

7.6. Переход от диагностики к восстановлению

Первичная диагностика ограничена фиксацией, просмотром и сопоставлением сведений. Перезапуск компонентов, удаление ресурсов, очистка кэша и ручная правка учёта не являются универсальными решениями.

Восстановительные изменения выполняет уполномоченный исполнитель по эксплуатационному регламенту. До изменения фиксируются цель, влияние на пользователей и данные, условия останова и возврата, необходимые копии и способ проверки результата. При неизвестном исходе предыдущей операции сначала устанавливается её состояние.

После восстановления проверяются затронутая функция, готовность зависимостей и сохранение ограничений доступа. Доступность процесса не заменяет согласованность продуктового и платформенного состояния. Исходный запрос не должен получать повторный учёт.

Для штатного отказа политики результатом разбора является установленное ограничение и его объяснение владельцу нагрузки, а не разрешение запроса. Для частично выполненной операции проверяется состояние объектов, а не только готовность процесса.

При отсутствии разрешённого сценария восстановления инцидент передаётся с собранной диагностикой. Полный сброс и удаление данных не используются вместо регламента.

8. Дополнительные возможности

8.1. Включённые подсистемы

Дополнительные подсистемы используются в пределах установленной конфигурации. К ним относятся защита содержимого, платформенное хранение диалогов, автомасштабирование и средства наблюдаемости. Отсутствие необязательной подсистемы не является отказом базовой платформы. Выключенная подсистема не выполняет свою функцию.

В эксплуатационной конфигурации фиксируются включённые компоненты, их версии, зависимости и ответственные за сопровождение. Изменения продуктовых настроек отделяются от обслуживания инфраструктуры и данных.

Продуктовый учёт не заменяется необязательным платформенным учётом. PostgreSQL для продуктового журнала и ValKey промышленного продуктового режима сохраняются независимо от включения дополнительных подсистем.

8.2. Защита содержимого

Guardrails — включаемый слой проверки, маскирования и блокировки содержимого на шлюзе моделей. По умолчанию он выключен и не изменяет запросы и ответы. Правила безопасности и ограничения приведены в разделе 5.5.

Политики привязываются к модели и/или клиенту. Для контроля применяется итоговый набор политик, сформированный оператором, а не только наличие ресурса политики. Обязательную политику клиент не может отключить параметром запроса. Режим только аудита не блокирует передачу содержимого.

Защита содержимого дополняет разграничение доступа и контроль маршрутов. Отсутствие срабатывания детектора не доказывает безопасность данных. Внешний проверяющий сервис является отдельным получателем данных и используется только в разрешённой схеме передачи.

8.3. Хранение диалогов

Платформенное хранение диалогов — отдельная необязательная подсистема. Содержимое диалога отличается от продуктового журнала потребления, аудита безопасности и локальной истории клиентского приложения.

Сохранение содержимого допускается только по разрешённой политике обработки данных. До включения хранения закрепляются цель, категории данных, места хранения, получатели, круг доступа, сроки и порядок удаления, включая резервные копии. Без этих условий сохранение содержимого для диагностики не включают.

Права просмотра аналитики не предоставляют права чтения диалогов. Аудит безопасности ведётся без ключей, тел запросов и текстов обращений к модели; включение хранения диалогов не отменяет этого ограничения.

Отключение сохранения новых диалогов и удаление накопленных данных — разные операции. Отключение доступа не является удалением данных. Правила обращения с данными приведены в разделе 6.7.

8.4. Автомасштабирование и наблюдаемость

Переход между нулём и одним экземпляром модели выполняет оператор. KEDA управляет масштабом от одного до нескольких экземпляров, используя метрики Prometheus или VictoriaMetrics. Работа без KEDA не означает наличие другого механизма автоматического масштабирования от одного до нескольких экземпляров.

При запуске из нулевого масштаба возникает холодный старт. Запрос удерживается в пределах отведённого времени ожидания либо получает HTTP 503. Этот код сам по себе не устанавливает причину отказа: его сопоставляют с состоянием модели.

Средства наблюдаемости предоставляют метрики оператора, шлюза и вычислительных ресурсов. Отсутствие панели или метрики не доказывает отказ модели. Для диагностики сопоставляются время, модель, включённая конфигурация и доступные сведения о состоянии. Порядок разбора приведён в главе 7.

8.5. Доступ и обслуживание дополнительных подсистем

Дополнительная подсистема не отменяет продуктовые проверки членства, подписки, разрешённых моделей, расписаний и квот. Сетевое соединение с внешним контуром не предоставляет право передавать туда любые данные.

Секреты подключений не включаются в описания, журналы и диагностические выгрузки. Для обращения в поддержку используются технические сведения; пользовательские документы и ответы моделей не заменяют диагностику.

Отключение дополнительной подсистемы выполняется с учётом её данных, зависимостей и активных обращений. Удаление ресурса или пакета не используется как

универсальный способ очистки связанных хранилищ. Ограничения обновления и вывода из эксплуатации приведены в разделах 6.8 и 6.9.