КОМПОНЕНТЫ И ПОТОКИ ДАННЫХ

Архитектура

Продукт: СОПКА.Рой · Версия: 1.0.0 · Агент: 1.0.1 · Протокол: 1 · Редакция: 15.10.2026

Правообладатель ПО: Общество с ограниченной ответственностью «Каннам». ИНН 2543020645, ОГРН 1132543001502. Реквизиты приведены в отдельном разделе документации.

Состав системы

СОПКА.Рой 1.0.0 включает веб-приложение, фоновые задачи, хранилища данных и Windows-агент. Пользователи и агенты подключаются к опубликованному HTTPS-адресу через reverse proxy; служебные хранилища не предназначены для прямого доступа извне.

Компонент	Ответственность
Nginx и приложение	Публикация веб-интерфейса/API, аутентификация, права workspace, маршрутизация Socket.IO.
PostgreSQL	Пользователи, рабочие области, клиенты, политики, инциденты, команды, каталог найденных файлов и metadata объектов.
ClickHouse	История DLP-событий и запросы журналов, счётчиков и отчётов.
RustFS / S3	Бинарные объекты: снимки, вложения и временные файловые материалы.
task-worker	Фоновые задачи, включая доставку событий из outbox в ClickHouse.
Аналитический контур	Сбор эксплуатационных показателей и read-only панели Grafana с отдельными правами.

Как событие попадает в журнал

1. Агент отправляет событие с идентификаторами устройства и рабочей области.

2. Сервер проверяет аутентификацию и сохраняет событие в PostgreSQL outbox вместе с квитанцией приёма.
3. Фоновая задача передаёт событие в ClickHouse.
4. После подтверждения ClickHouse запись выводится из очереди доставки; основной журнал и поиск читают подтверждённую историю из ClickHouse. Список снимков и получение отдельного только что принятого события могут учитывать ожидающие записи до переноса; это не вторая постоянная база журналов.

Наличие строки dlp_events в PostgreSQL не означает вторую постоянную базу журналов: это очередь доставки. При задержке outbox событие может быть принято сервером, но ещё не появиться в журнале. Отсутствие новых строк в интерфейсе требует проверки worker и ClickHouse, а не немедленной переустановки агента.

Файлы и объектное хранилище

PostgreSQL хранит metadata, статус и принадлежность объекта рабочей области; содержимое находится в S3-совместимом хранилище. Ссылка на metadata не заменяет резервную копию самого объекта. Доступ к материалам контролируется приложением.

Агент Windows

Служба CopkaRoyAgentService отвечает за идентичность устройства, heartbeat, политики и фоновые операции. CopkaRoyUserAgent работает в интерактивной сессии и обеспечивает функции рабочего стола. Работа службы без пользовательского процесса не подтверждает готовность снимков, прямого экрана или PowerShell.

Команды и оперативный обмен

Агент регулярно сообщает состояние и получает ожидающие команды. Для оперативного обмена может создаваться временное соединение. Постановка, доставка, начало и завершение команды являются отдельными этапами. Несколько экземпляров сервера используют согласованное оперативное состояние; маршрутизация интерактивных соединений зависит от схемы развёртывания.

Изоляция рабочих областей

Сервер определяет workspace по проверенной сессии пользователя или identity агента. Параметры из формы и JSON не дают доступа к чужому клиенту. Аналитические панели владельца инсталляции являются отдельным административным контуром, а не пользовательским доступом ко всем компаниям.

Миграции и запуск

Структура PostgreSQL управляется Alembic отдельным этапом db-init. Мигратор и прикладной пользователь БД имеют разные полномочия; обычный web-процесс не должен создавать таблицы. При обновлении запись останавливают на предусмотренный процедурой период, а приложение запускают после успешных миграций.

Время и идентификаторы

Сетевые временные метки передаются в UTC, интерфейс показывает выбранный часовой пояс. UUID клиента, идентификатор события и идентификатор команды относятся к разным сущностям; в диагностике сохраняйте их вместе с workspace и временем операции.

Границы версии 1.0.0

Поставляемый агент предназначен для Windows x64. Linux-агент в состав версии 1.0.0 не входит. Прямой экран предназначен для просмотра; удалённый ввод мыши и клавиатуры в текущем интерфейсе не предоставляется. Поддержка конкретного действия дополнительно определяется capabilities и правами оператора.

Оценка политик и масштабирование

Правила оцениваются агентом в пределах платформы, разрешённых функций и области политики. Сначала учитывается приоритет политики, затем правила; критичность относится к результату. Исключения действуют на выбранные операции, а факт совпадения отличается от факта исполнения. Сервер хранит и отображает присланные результаты и управляет командами. Параллельная обработка обеспечивается несколькими экземплярами серверного компонента. Согласованность оперативного состояния и управление соединениями обеспечиваются эксплуатационными компонентами поставки. Ресурсный профиль на 1000 агентов приведён в «Системных требованиях» и требует отдельного измерения.

Архитектура AI-чата

Flask обрабатывает приватную историю и NDJSON-поток ответа. PostgreSQL хранит диалоги, сообщения, квоты и аренды допуска; RLS изолирует переписки по пользователю и области. Переключение диалогов сохраняет историю и выбранную рабочую область. Модель обращается к внутреннему диспетчеру MCP для ограниченных выборок; события берутся из ClickHouse.

Подробнее: AI-чат: анализ безопасности.