### **Таблица** apps **(Основная таблица приложений)**

| **Столбец** | **Тип** | **Обязательное** | **Описание** | **Обоснование добавления** |
| --- | --- | --- | --- | --- |
| id | UUID | Да | Уникальный идентификатор приложения | Стандарт для микросервисов, гарантирует глобальную уникальность |
| name | VARCHAR(255) | Да | Уникальное имя приложения (например, billing-service) | Человекочитаемый идентификатор, используется для поиска |
| description | TEXT | Нет | Описание назначения приложения | Для документации и управления |
| secret\_key | BYTEA | Да | Зашифрованный ключ приложения (AES-256) | Безопасное хранение секретов. BYTEA позволяет хранить бинарные данные |
| is\_active | BOOLEAN | Нет (DEFAULT TRUE) | Флаг активности приложения | Мягкое удаление/блокировка приложения без физического удаления данных |
| created\_at | TIMESTAMPTZ | Да (DEFAULT NOW()) | Дата создания записи | Аудит и анализ временных меток |
| updated\_at | TIMESTAMPTZ | Да (DEFAULT NOW()) | Дата последнего обновления | Контроль изменений |
| created\_by | UUID | Да | ID пользователя, создавшего запись (из сервиса users) | Аудит действий. Ссылка на внешнюю систему |
| metadata | JSONB | Нет (DEFAULT '{}') | Дополнительные метаданные (версия, окружение, теги) | Гибкость для хранения неструктурированных данных без изменения схемы |

**Индексы:**

* idx\_apps\_name – Ускорение поиска по имени (частая операция).
* idx\_apps\_created\_by – Оптимизация запросов по создателю.

### **Таблица =** key\_rotation\_history **(История ротации ключей)**

| **Столбец** | **Тип** | **Обязательное** | **Описание** | **Обоснование добавления** |
| --- | --- | --- | --- | --- |
| id | UUID | Да | Уникальный идентификатор записи | Стандартная практика для PK |
| app\_id | UUID | Да | Ссылка на приложение (apps.id) | Связь с основной таблицей. ON DELETE CASCADE автоматически удаляет историю при удалении приложения |
| old\_key | BYTEA | Нет | Предыдущий зашифрованный ключ | Для аудита и отката изменений. Может быть NULL при первой генерации |
| new\_key | BYTEA | Да | Новый зашифрованный ключ | Фиксация нового значения |
| rotated\_at | TIMESTAMPTZ | Да (DEFAULT NOW()) | Дата ротации | Анализ временных меток |
| rotated\_by | UUID | Да | ID пользователя, выполнившего ротацию (из сервиса users) | Аудит действий |

**Индекс:**

* idx\_key\_rotation\_app\_id – Ускорение выборки истории для конкретного приложения.

### **Ключевые решения и их обоснование**

1. **Типы данных:**
   * UUID **вместо** INTEGER:
     + Глобальная уникальность (важно для распределенных систем).
     + Защита от перебора (нельзя угадать ID).
   * BYTEA **для ключей**:
     + Хранение зашифрованных данных (AES-256 требует бинарного формата).
     + Безопаснее, чем TEXT (нет риска кодирования/декодирования).
   * JSONB **для** metadata:
     + Поддержка индексов и сложных запросов (в отличие от JSON).
     + Эффективное хранение и обработка.
2. **Аудит и безопасность:**
   * created\_by**/**rotated\_by:
     + Отслеживание действий пользователей (требование compliance).
     + Связь с сервисом users через UUID.
   * is\_active **вместо удаления**:
     + Сохранение истории без физического удаления данных.
     + Возможность восстановления.
3. **Производительность:**
   * **Индексы на** name **и** created\_by:
     + Частые запросы по этим полям (поиск приложений по имени или создателю).
   * **Отдельная таблица для истории ключей**:
     + Избегание "раздувания" основной таблицы.
     + Оптимизация запросов к актуальным данным.
4. **Гибкость:**
   * metadata **(JSONB)**:
     + Возможность хранить:

{

"environment": "production",

"version": "2.1.0",

"team": "devops"

}

* + - Без изменения схемы БД.

### **Примеры использования полей**

1. **Создание приложения:**

INSERT INTO apps (name, description, created\_by, secret\_key)

VALUES (

'payment-service',

'Обработка платежей',

'a0eebc99-9c0b-4ef8-bb6d-6bb9bd380a11',

encrypt('new-secret-key', 'master-key')

);

1. **Ротация ключа:**

-- 1. Получаем старый ключ

-- 2. Генерируем новый

-- 3. Обновляем запись в apps

-- 4. Фиксируем в истории:

INSERT INTO key\_rotation\_history (app\_id, old\_key, new\_key, rotated\_by)

VALUES (

'123e4567-e89b-12d3-a456-426614174000',

encrypt('old-key', 'master-key'),

encrypt('new-key', 'master-key'),

'a0eebc99-9c0b-4ef8-bb6d-6bb9bd380a11'

);

1. **Запрос метаданных:**

SELECT name, metadata->>'environment'

FROM apps

WHERE metadata->>'team' = 'devops';

### **Что можно улучшить в будущем?**

1. **Шифрование**:
   * Интеграция с KMS (например, AWS KMS или HashiCorp Vault).
   * Использование pgcrypto для шифрования внутри БД.
2. **Роли и доступы**:
   * Добавление таблицы app\_permissions для тонкого контроля.
3. **Временные метки с часовыми поясами**:
   * TIMESTAMPTZ уже учитывает это, но можно добавить явное указание зоны.

Эта структура обеспечивает баланс между:

* **Безопасностью** (шифрование, аудит).
* **Гибкостью** (JSONB, мягкое удаление).
* **Производительностью** (индексы, нормализация).