

## # Войслаб: Партнёрское API

Документ описывает публичный контракт Партнёрского API Войслаб для интеграции партнёрских систем с личным кабинетом и клиентскими организациями.

## Назначение

Партнёрское API позволяет партнёру управлять уже созданными клиентскими организациями Войслаб через machine-to-machine интеграцию:

- получать список своих клиентских организаций;
- получать доступные голоса Робота для клиентской организации;
- создавать и обновлять роботов;
- получать номер и добавочный для тестового звонка Роботу;
- создавать и запускать базы контактов;
- получать входящий номер базы контактов;
- добавлять контакты;
- получать статусы контактов и прогресс базы;
- включать webhook-уведомления базы контактов после звонков.

Создание клиентских организаций через API не поддерживается.

## Базовый URL

```
https://voicelabs.ru/api/v1/partner-api
```

## Где Настраивается API

Партнёр сам создаёт API keys в личном кабинете Войслаб в разделе **Партнёры**, вкладка **API**.

Вкладка **API** доступна владельцам и редакторам партнёрской организации. В ней доступны:

- API keys;
- текущие лимиты API;
- ссылка на эту документацию.

Webhook-уведомления настраиваются не глобально, а на уровне конкретной базы контактов через API базы.

Лимиты API задаются администратором Войслаб и отображаются партнёру только для чтения.

## Авторизация

Все запросы выполняются с API key:

```
Authorization: Bearer vlp_live_XXXXXXXXXXXXXXXXXXXX
```

API key привязан к одной партнёрской организации. Партнёр видит только клиентские организации, которые привязаны к этой партнёрской организации.

Если ключ отозван, уже начатый запрос может завершиться, но следующие запросы с этим ключом получают ошибку авторизации.

## Лимиты

Значения по умолчанию:

Лимит	Значение
Роботы	100 на каждую клиентскую организацию
Базы контактов	100 на каждую клиентскую организацию
Контакты	5000 в каждой базе контактов
Batch-добавление контактов	до 1000 контактов в одном запросе

Rate limit по умолчанию:

Тип запросов	Лимит
Read-запросы	60 запросов в минуту
Write-запросы	20 запросов в минуту
Batch-добавление контактов	5 запросов в минуту
Настройка уведомлений	20 запросов в минуту

## Дубли Контактв

Для контактов действует защита от дублей:

- если телефон уже есть в этой базе, строка будет пропущена с ошибкой `PHONE_DUPLICATE_IN_LIST` ;
- если телефон уже есть в другой базе этой клиентской организации, строка будет пропущена с ошибкой `PHONE_DUPLICATE_CROSS_LIST` ;
- batch-запрос может быть частично успешным: валидные новые контакты создаются, проблемные строки возвращаются в `errors` .

Batch-запрос безопасно повторить при сетевой ошибке: уже созданные контакты будут отсеяны как дубли, а новые валидные строки будут добавлены.

## Формат Ошибок

Ошибки возвращаются простым JSON-объектом:

```
{
  "errorCode": "PHONE_INVALID",
  "message": "Неверный формат телефона",
  "traceId": "00-...",
  "details": {
    "field": "phoneE164"
  }
}
```

HTTP status остаётся основным транспортным статусом ( 400 , 401 , 403 , 404 , 409 , 429 , 500 ), а `errorCode` нужен для программной обработки.

Основные коды ошибок:

Код	Значение
<code>PARTNER_API_KEY_INVALID</code>	API key отсутствует или неверный.
<code>PARTNER_API_KEY_REVOKED</code>	API key отозван.
<code>PARTNER_API_DISABLED</code>	Партнёрское API выключено для партнёрской организации.
<code>INSUFFICIENT_SCOPE</code>	У ключа нет нужного права.
<code>CLIENT_ORG_NOT_FOUND</code>	Клиентская организация не найдена или не принадлежит партнёру.
<code>RESOURCE_NOT_FOUND</code>	Робот, база контактов, контакт или webhook-уведомление не найдено в доступной клиентской организации.
<code>CLIENT_ORG_INACTIVE</code>	Клиентская организация выключена.
<code>CLIENT_ORG_NOT_VERIFIED</code>	Организация не верифицирована для запуска базы.
<code>INSUFFICIENT_BALANCE</code>	Нельзя активировать базу из-за нулевого или отрицательного баланса.
<code>CALL_LIST_HAS_NO_ROBOT</code>	Нельзя активировать базу без робота.
<code>CALL_LIST_NO_PENDING_CONTACTS</code>	Нельзя активировать базу без контактов в ожидании звонка.
<code>ROBOT_NO_DEFAULT_VOICE</code>	Нельзя создать робота: у клиентской организации нет доступного голоса для автоматического выбора.
<code>ROBOT_VOICE_NOT_AVAILABLE</code>	Переданный <code>voiceConfigId</code> не доступен клиентской организации.
<code>ROBOT_LIMIT_EXCEEDED</code>	Превышен лимит роботов.
<code>CALL_LIST_LIMIT_EXCEEDED</code>	Превышен лимит баз контактов.
<code>CONTACT_LIMIT_EXCEEDED</code>	Превышен лимит контактов в базе.
<code>CONTACT_LIST_TOO_LARGE_FOR_FULL_EXPORT</code>	База слишком большая для полной выгрузки контактов.
<code>EXTRA_INVALID</code>	<code>extra</code> содержит неподдерживаемый формат.
<code>EXTRA_TOO_LARGE</code>	<code>extra</code> превышает лимит размера или количества полей.
<code>PHONE_INVALID</code>	Неверный формат телефона.
<code>PHONE_BLACKLISTED</code>	Телефон в чёрном списке.

Код	Значение
PHONE_DUPLICATE_IN_LIST	Телефон уже есть в этой базе.
PHONE_DUPLICATE_CROSS_LIST	Телефон уже есть в другой базе этой организации.
NOTIFICATION_INVALID_URL	Некорректный URL webhook-уведомления.
NOTIFICATION_TEST_FAILED	Тестовая отправка webhook-уведомления не удалась.
RATE_LIMIT_EXCEEDED	Превышен лимит запросов.
CONCURRENCY_CONFLICT	Параллельная операция с тем же лимитом уже выполняется, повторите запрос позже.

## Семантика PATCH

В `PATCH` передаются только поля, которые нужно изменить.

- Если поле не передано, оно не меняется.
- Если nullable-поле передано со значением `null`, оно сбрасывается.

Пример сброса лимита:

```
{
  "totalCallLimit": null
}
```

Такой запрос снимет общий лимит звонков. Если поле `totalCallLimit` не передавать вообще, текущий лимит останется без изменений.

## Health

Проверка доступности API и валидности ключа.

```
GET /health
```

Ответ:

```
{
  "status": "ok",
  "partnerOrgId": "guid",
  "partnerName": "Партнёр",
  "keyId": "guid",
  "keyName": "Основной ключ",
  "timestampUtc": "2026-04-27T10:00:00Z"
}
```

## Клиентские Организации

### Список Организаций

```
GET /client-orgs
```

Ответ:

```
[
  {
    "id": "guid",
    "name": "000 Клиент",
    "isActive": true,
    "isVerified": true,
    "balanceCents": 123000,
    "createdAtUtc": "2026-04-27T10:00:00Z"
  }
]
```

`balanceCents` возвращается намеренно: партнёр может показывать в своей системе, достаточно ли средств у клиентской организации для запуска обзвона.

### Одна Организация

```
GET /client-orgs/{orgId}
```

## Голоса Робота

Партнёр может получить список голосов, доступных конкретной клиентской организации, и передать выбранный `voiceConfigId` при создании или обновлении Робота.

```
GET /client-orgs/{orgId}/voices
```

Ответ:

```
[
  {
    "id": "guid",
    "name": "Дмитрий",
    "description": "Мужской голос"
  }
]
```

В ответе нет внутренних настроек синтеза речи и внешних vendor-полей. Возвращаются только включённые публичные голоса и приватные голоса, назначенные этой клиентской организации.

# Роботы

## Список Роботов

```
GET /client-orgs/{orgId}/robots
```

## Создать Робота

```
POST /client-orgs/{orgId}/robots
```

`voiceConfigId` можно не передавать: тогда система выберет доступный голос автоматически — сначала системный голос по умолчанию, а если он недоступен клиентской организации, первый доступный `enabled voice`. Если партнёр хочет выбрать голос явно, сначала получите список доступных голосов через `GET /client-orgs/{orgId}/voices`.

Если у клиентской организации нет ни одного доступного голоса для автоматического выбора, API вернёт `ROBOT_NO_DEFAULT_VOICE`.

Если переданный `voiceConfigId` не входит в список доступных голосов организации или голос выключен, API вернёт `ROBOT_VOICE_NOT_AVAILABLE`.

Ограничения полей робота совпадают с личным кабинетом:

Поле	Ограничение
<code>name</code>	до 100 символов
<code>voiceConfigId</code>	<code>guid</code> доступного голоса или <code>null</code> для автоматического выбора
<code>sellerRole</code> , <code>speechStyle</code> , <code>conversationGoal</code>	до 500 символов
<code>goalCriteria</code> , <code>rules</code> , <code>argumentsAndFacts</code>	до 10 элементов, каждый до 500 символов
<code>conversationStart</code>	до 3 элементов, каждый до 500 символов
<code>conversationPlan</code>	от 1 до 10 элементов, каждый до 1000 символов

Запрос:

```
{
  "name": "Обзвон клиентов",
  "voiceConfigId": "guid",
  "sellerRole": "Менеджер отдела продаж",
  "speechStyle": "Дружелюбно, уверенно, без давления",
  "conversationGoal": "Выявить интерес к продукту и договориться о следующем шаге",
  "goalCriteria": [
    "Клиент согласился на встречу",
    "Клиент попросил отправить коммерческое предложение"
  ],
  "rules": [
    "Не спорить с клиентом",
    "Если клиенту неудобно говорить, предложить перезвонить"
  ],
  "argumentsAndFacts": [
    "Компания работает с 2020 года",
    "Есть интеграция с CRM"
  ],
  "conversationStart": [
    "Здравствуйте, удобно сейчас пару минут?"
  ],
  "conversationPlan": [
    "Коротко представиться",
    "Уточнить текущую задачу клиента",
    "Предложить следующий шаг"
  ]
}
```

Ответы `GET`, `POST` и `PATCH` по Роботу содержат read-only поле `callNumber`:

```

{
  "id": "guid",
  "organizationId": "guid",
  "name": "Обзвон клиентов",
  "voiceConfigId": "guid",
  "callNumber": {
    "phoneNumber": "89846609488",
    "extension": "61788",
    "dialString": "89846609488,61788"
  },
  "sellerRole": "Менеджер отдела продаж",
  "speechStyle": "Дружелюбно, уверенно, без давления",
  "conversationGoal": "Выявить интерес к продукту и договориться о следующем шаге",
  "goalCriteria": [],
  "rules": [],
  "argumentsAndFacts": [],
  "conversationStart": [],
  "conversationPlan": [
    "Коротко представиться",
    "Уточнить текущую задачу клиента",
    "Предложить следующий шаг"
  ],
  "createdAtUtc": "2026-06-10T10:00:00Z",
  "updatedAtUtc": "2026-06-10T10:00:00Z"
}

```

`callNumber` нужен, чтобы партнёрская система могла показать пользователю, как позвонить Роботу для тестового входящего звонка. Поля `phoneNumber`, `extension` и `dialString` нельзя менять через партнёрское API. Если системный номер для входящих тестовых звонков не настроен, все три поля будут `null`.

## Получить Робота

```
GET /client-orgs/{orgId}/robots/{robotId}
```

## Обновить Робота

```
PATCH /client-orgs/{orgId}/robots/{robotId}
```

В `PATCH` передаются только поля, которые нужно изменить. API принимает только поля сценария робота и `voiceConfigId`, перечисленные выше. Если передать `"voiceConfigId": null`, Робот вернётся к автоматическому выбору доступного голоса.

При создании Робота `conversationPlan` обязателен и должен содержать хотя бы один непустой пункт. При `PATCH`, если меняется любое поле сценария Робота, итоговый `conversationPlan` также должен содержать хотя бы один пункт. Для старых Роботов с пустым планом чтение и `PATCH` только служебных полей, например `name` или `voiceConfigId`, остаются совместимыми.

# Базы Контактов

## Список Баз

```
GET /client-orgs/{orgId}/call-lists
```

Ответы `GET`, `POST`, `PATCH`, `activate` и `deactivate` по базе содержат read-only поля `robotCallNumber` и `inboundPhoneNumber`:

```
{
  "id": "guid",
  "organizationId": "guid",
  "name": "Апрельская база",
  "robotId": "guid",
  "robotName": "Обзвон клиентов",
  "robotCallNumber": {
    "phoneNumber": "89846609488",
    "extension": "61788",
    "dialString": "89846609488,61788"
  },
  "inboundPhoneNumber": "+79617541756",
  "status": "Draft",
  "isActive": false,
  "schedule": {
    "workingDays": 31,
    "workingHoursStart": "09:00",
    "workingHoursEnd": "18:00",
    "timeZoneId": "Europe/Moscow"
  },
  "totalCallLimit": 1000,
  "totalCallLimitCallsMade": 0,
  "goalAchievedLimit": 100,
  "goalAchievedLimitCallsMade": 0,
  "createdAtUtc": "2026-06-10T10:00:00Z",
  "updatedAtUtc": "2026-06-10T10:00:00Z"
}
```

`robotCallNumber` совпадает с номером для кнопки «Позвонить Роботу» у Робота, привязанного к базе.

`inboundPhoneNumber` — входящий номер вкладки «Входящие» у базы контактов. Оба поля доступны только для чтения и не принимаются в `POST` / `PATCH`.

## Создать Базу

```
POST /client-orgs/{orgId}/call-lists
```

Запрос:

```
{
  "name": "Апрельская база",
  "robotId": "guid",
  "schedule": {
    "workingDays": 31,
    "workingHoursStart": "09:00",
    "workingHoursEnd": "18:00",
    "timeZoneId": "Europe/Moscow"
  },
  "totalCallLimit": 1000,
  "goalAchievedLimit": 100
}
```

`schedule`, `totalCallLimit`, `goalAchievedLimit` можно не передавать.

`totalCallLimit` — общий лимит звонков по базе.

`goalAchievedLimit` — отдельный лимит звонков со статусом “цель достигнута”.

Если любой из лимитов достигнут, база автоматически выключается.

После автоотключения оба лимита и оба счётчика лимитов сбрасываются. Следующий запуск базы начинается с новой настройки лимитов.

## Расписание И `workingDays`

`workingDays` — сумма значений дней недели.

День	Значение
Понедельник	1
Вторник	2
Среда	4
Четверг	8
Пятница	16
Суббота	32
Воскресенье	64

Примеры:

Расписание	Значение
Будни, понедельник-пятница	31
Выходные, суббота-воскресенье	96
Каждый день	127

$31 = 1 + 2 + 4 + 8 + 16$ , то есть понедельник-пятница.

Значение `0` означает, что робот не звонит ни в один день.

## Получить Базу

```
GET /client-orgs/{orgId}/call-lists/{callListId}
```

## Обновить Базу

```
PATCH /client-orgs/{orgId}/call-lists/{callListId}
```

Можно менять:

- `name`;
- `schedule`;
- `totalCallLimit`;
- `goalAchievedLimit`;
- `robotId`, только если база выключена.

Нельзя менять:

- телефонию;
- `robotCallNumber` и `inboundPhoneNumber`;
- CRM и уведомления;
- архивный или удалённый статус напрямую.

## Включить Базу

```
POST /client-orgs/{orgId}/call-lists/{callListId}/activate
```

Для включения через партнёрское API база должна принадлежать активной и верифицированной клиентской организации, иметь положительный баланс, работа и хотя бы один незаблокированный контакт в ожидании звонка.

## Отключить Базу

```
POST /client-orgs/{orgId}/call-lists/{callListId}/deactivate
```

## Прогресс Базы

```
GET /client-orgs/{orgId}/call-lists/{callListId}/progress
```

Ответ:

```
{
  "callListId": "guid",
  "totalContacts": 1000,
  "pending": 700,
  "inProgress": 5,
  "completed": 295,
  "goalAchieved": 80,
  "goalNotAchieved": 215,
  "blocked": 10,
  "totalCallLimit": 1000,
  "totalCallLimitCallsMade": 250,
  "goalAchievedLimit": 100,
  "goalAchievedLimitCallsMade": 80
}
```

## Контакты

### Получить Все Контакты Базы

```
GET /client-orgs/{orgId}/call-lists/{callListId}/contacts
```

Контакты возвращаются одним списком без фильтров и пагинации. Размер базы ограничен лимитом контактов.

`maxAttempts` показывает, сколько попыток дозвона система может сделать по контакту в текущей политике обзвона. Через партнёрское API это значение нельзя менять.

Ответ:

```
[
  {
    "id": "guid",
    "phoneE164": "+79001234567",
    "displayName": "Иван",
    "extra": {
      "city": "Москва"
    },
    "status": "Pending",
    "attemptCount": 1,
    "maxAttempts": 3,
    "attempts": [
      {
        "number": 1,
        "status": "NoAnswer",
        "durationSec": 0,
        "startedAtUtc": "2026-04-27T10:00:00Z",
        "callId": "guid"
      }
    ],
    "nextCallAtUtc": "2026-04-27T12:00:00Z",
    "lastAttemptAtUtc": "2026-04-27T10:00:00Z",
    "blockedAtUtc": null,
    "createdAtUtc": "2026-04-27T09:00:00Z"
  }
]
```

## Добавить Контакт

POST /client-orgs/{orgId}/call-lists/{callListId}/contacts

`extra` — дополнительные поля контакта. Правила:

- максимум `50` ключей;
- максимум `8 KB` после сериализации;
- значения могут быть только строками, числами или `boolean`;
- вложенные объекты и массивы не поддерживаются;
- служебные поля телефона и имени нужно передавать через `phoneE164` и `displayName`, а не дублировать в `extra`;
- не используйте ключи, начинающиеся с `_`: они зарезервированы для возможных системных полей;
- при нарушении формата вернётся `EXTRA_INVALID` или `EXTRA_TOO_LARGE`.

Запрос:

```
{
  "phoneE164": "+79001234567",
  "displayName": "Иван",
  "extra": {
    "city": "Москва",
    "budget": "100000"
  }
}
```

## Добавить Контакты Batch-Запросом

```
POST /client-orgs/{orgId}/call-lists/{callListId}/contacts/batch
```

Запрос:

```
{
  "contacts": [
    {
      "phoneE164": "+79001234567",
      "displayName": "Иван",
      "extra": {
        "city": "Москва"
      }
    }
  ]
}
```

Ответ может быть частично успешным:

```
{
  "total": 10,
  "created": 8,
  "skipped": 2,
  "items": [
    {
      "index": 0,
      "contactId": "guid",
      "phoneE164": "+79001234567",
      "status": "Created"
    }
  ],
  "errors": [
    {
      "index": 3,
      "phoneE164": "123",
      "errorCode": "PHONE_INVALID",
      "reason": "Неверный формат телефона"
    }
  ]
}
```

## Получить Контакт

```
GET /client-orgs/{orgId}/call-lists/{callListId}/contacts/{contactId}
```

## Заблокировать Контакт

```
POST /client-orgs/{orgId}/call-lists/{callListId}/contacts/{contactId}/block
```

## Разблокировать Контакт

```
POST /client-orgs/{orgId}/call-lists/{callListId}/contacts/{contactId}/unlock
```

## Webhook-Уведомления Базы

Webhook-уведомления используют существующую систему уведомлений базы контактов. Партнёр не создаёт отдельный глобальный webhook, а включает webhook-уведомление для конкретной базы контактов.

### Список Webhook-Уведомлений Базы

```
GET /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook
```

### Создать Webhook-Уведомление

```
POST /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook
```

Запрос:

```
{
  "webhookUrl": "https://partner.example.com/voicelab/webhook",
  "scope": "All",
  "description": "Основной webhook партнёра",
  "includeTranscript": true,
  "isActive": true
}
```

Если `includeTranscript = true`, в webhook будет передаваться текст разговора. Такой текст может содержать персональные данные, поэтому партнёр отвечает за защиту endpoint-а, логов и хранилищ, куда он принимает payload.

`webhookUrl` должен быть публичным `https://` URL. `http://`, `localhost`, служебные, локальные и недоступные из публичного интернета IP-адреса, а также DNS-имена, которые указывают на такие адреса, не принимаются.

`scope` :

Scope	Когда отправлять
<code>LeadsOnly</code>	Только звонки со статусом <code>GoalAchieved</code> .
<code>AllSuccessful</code>	Состоявшиеся разговоры: <code>GoalAchieved</code> , <code>CallLater</code> , <code>NotInterested</code> , <code>Negative</code> , <code>HungUp</code> . Старое значение сохранено для совместимости.

Scope	Когда отправлять
All	Все завершённые звонки: GoalAchieved, CallLater, NotInterested, Negative, HungUp, NoAnswer, Voicemail, Silence, Error.

Для сценария “после каждого звонка” используйте `All`. Технический промежуточный статус `Initiated` в webhook не отправляется.

## Получить Webhook-Уведомление

```
GET /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook/{notificationId}
```

## Обновить Webhook-Уведомление

```
PATCH /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook/{notificationId}
```

Можно менять `webhookUrl`, `scope`, `description`, `includeTranscript`, `isActive`.

Webhook secret при `PATCH` не меняется. Если `includeTranscript` переключить на `true`, текст разговора начнёт приходить только в следующих webhook.

## Включить Или Отключить Webhook-Уведомление

```
POST /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook/{notificationId}/toggle
```

Запрос:

```
{
  "isActive": true
}
```

## Тест Webhook-Уведомления

```
POST /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook/{notificationId}/test
```

Тестовая отправка использует фиктивный payload и отдельный event header:

```
X-Webhook-Event: notification.test
```

Тест не создаёт реальный звонок и не является боевым уведомлением. Если endpoint партнёра вернул неуспешный HTTP-статус, не ответил за 5 секунд или произошла ошибка DNS/TLS/соединения, API вернёт

```
NOTIFICATION_TEST_FAILED
```

## Удалить Webhook-Уведомление

```
DELETE /client-orgs/{orgId}/call-lists/{callListId}/notifications/webhook/{notificationId}
```

Удаление убирает настройку webhook-уведомления и pending-задачи доставки по нему. Используйте это, если секрет потерян и нужно создать webhook заново.

## Payload

Webhook payload содержит:

- `callId` ;
- `direction` ( `outbound` или `inbound` , добавлено с релиза 2026-05-25);
- `organizationName` ;
- `callListId` ;
- `callListName` ;
- `callListItemId` ;
- `phoneE164` ;
- `contactName` ;
- `status` ;
- `isGoalAchieved` ;
- `goalSummary` ;
- `goalConversationResume` ;
- `durationSeconds` ;
- `recordingUrl` ;
- `calledAtUtc` ;
- `contactFields` ;
- `transcript` , если `includeTranscript = true` .

`recordingUrl` может быть `null` , если запись недоступна или ещё не обработана. Для входящих звонков `callListItemId` и `contactName` обычно равны `null` , потому что входящий звонок привязывается к Базе контактов без создания контакта.

Значения `status` :

Status	Значение
<code>GoalAchieved</code>	Цель звонка достигнута.
<code>CallLater</code>	Клиент попросил перезвонить.
<code>NotInterested</code>	Клиенту не интересно.
<code>Negative</code>	Негатив, запрет контакта или похожий жёсткий отказ.
<code>HungUp</code>	Клиент сбросил звонок.
<code>NoAnswer</code>	Клиент не ответил.
<code>Voicemail</code>	Сработал автоответчик.
<code>Silence</code>	Клиент молчал или не дал распознаваемого ответа.
<code>Error</code>	Звонок завершился ошибкой обработки.

`Initiated` — технический промежуточный статус активного звонка; в webhook payload он не ожидается.

Пример payload:

```
{
  "callId": "11111111-1111-1111-1111-111111111111",
  "direction": "outbound",
  "organizationName": "Клиент партнёра",
  "callListId": "22222222-2222-2222-2222-222222222222",
  "callListName": "Апрельская база",
  "callListItemId": "33333333-3333-3333-3333-333333333333",
  "phoneE164": "+79001234567",
  "contactName": "Иван",
  "status": "GoalAchieved",
  "isGoalAchieved": true,
  "goalSummary": "Клиент заинтересовался и попросил связаться с менеджером.",
  "goalConversationResume": "Клиент подтвердил интерес, уточнил условия и согласился на следующий контакт.",
  "durationSeconds": 146,
  "recordingUrl": "https://voicelabs.ru/public/recordings/11111111-1111-1111-1111-111111111111.wav",
  "calledAtUtc": "2026-04-27T10:15:00Z",
  "contactFields": {
    "city": "Москва",
    "source": "landing"
  },
  "transcript": "Робот: Добрый день... Клиент: Да, интересно..."
}
```

## Подпись

Webhook secret нужен, чтобы партнёр мог проверить, что webhook действительно отправил Войслаб.

При создании webhook-уведомления сервер генерирует секрет и возвращает его в ответе только один раз:

```
{
  "id": "guid",
  "webhookUrl": "https://partner.example.com/voicelab/webhook",
  "scope": "All",
  "includeTranscript": true,
  "isActive": true,
  "webhookSecret": "whsec_XXXXXXXXXXXXXXXXXXXX"
}
```

Партнёр сохраняет `webhookSecret` у себя и использует его для проверки входящих webhook-запросов.

Каждая доставка содержит headers:

```
X-Webhook-Event: lead.goal_achieved | call.completed
X-Webhook-Id: guid
X-Webhook-Signature: sha256=...
```

Для `GoalAchieved` заголовок `X-Webhook-Event` равен `lead.goal_achieved`; для остальных завершённых статусов — `call.completed`. Тестовая отправка из Partner API использует `notification.test`.

Подпись считается так:

```
signature = HMAC-SHA256(raw_json_payload, webhookSecret)
```

Проверка на стороне партнёра:

1. Получить raw body webhook-запроса без изменения JSON.
2. Посчитать `hmac-sha256(raw body, webhookSecret)`.
3. Сравнить результат с `X-Webhook-Signature`.
4. Если подпись не совпала, webhook нужно отклонить.

Важно использовать именно raw body запроса. Если сначала распарсить JSON и собрать его заново, порядок пробелы могут измениться, и подпись не совпадёт.

Отдельного endpoint-а для смены секрета нет.

Если секрет потерян, удалите webhook-уведомление и создайте новое.

---

## Ограничения

- Создание клиентских организаций через API.
- История звонков через pull API.
- Назначение или изменение номера для тестового звонка Роботу.
- Назначение или изменение входящего номера базы контактов.
- Изменение настроек, которые не описаны в этом документе.
- Балансные операции.
- Ручной сброс статуса контакта.
- Ручной запуск звонка по контакту.
- Удаление организаций, роботов, баз или контактов.