

«10» февраля 2026 г.

Инструкция по получению доступа и установке «taskITnow»

Данная инструкция описывает **алгоритм для получения доступа к ПО TaskitNow, развернутому в публичном облаке**, по модели подписки (SaaS).

1. Необходимо перейти в браузере по адресу <https://taskitnow.ru/login>.
2. В зависимости от наличия ранее созданной учетной записи следует выбрать соответствующий таб – «Вход» или «Регистрация» (для создания записи). Примечание. При авторизации посредством социальных сетей или SSO – выбирайте соответствующую кнопку на страничке логина.
3. После ввода почты, являющейся идентификатором простой учетной записи, и пароля, нажмите кнопку «Войти» для запуска процедуры логина.

Входные данные на сервер с предустановленным ПО TaskITnow

| | |
|-------------------------|--|
| URL | https://taskitnow.ru |
| Логин/электронная почта | info@taskitnow.ru |
| Пароль | |

Для получения доступа к дистрибутиву ПО TaskitNow, предназначенному для развертывания в локальной инфраструктуре (on-premise), напишите запрос по адресу hello@taskitnow.ru.

Данная инструкция описывает **алгоритм установки ПО TaskitNow из предоставленного дистрибутива**.

Содержание

- [Установка из архива](#) (основной путь)
- [Управление](#)
- [Конфигурация](#)
- [SSL-сертификаты](#)
- [Устранение неполадок](#)
- [Сборка из исходников](#) (для разработчиков)
- [Создание дистрибутива](#) (для упаковки)
- [Архитектура](#)
- [Состав сервисов](#)

Системные требования

Для установки из архива

| Компонент | Минимум | Рекомендуется |
|----------------|---------------|-----------------------|
| ОС | Linux / macOS | Linux (Ubuntu 22.04+) |
| Docker | 24.0+ | 25.0+ |
| Docker Compose | v2 (плагин) | v2.20+ |
| RAM | 8 GB | 16 GB |
| Диск | 30 GB | 50 GB |

Дополнительно для сборки из исходников

| Компонент | Минимум | Рекомендуется |
|-----------|---------|---------------|
| Node.js | 22+ | 22 LTS |

| Компонент | Минимум | Рекомендуется |
|-----------|---------|---------------|
| npm | 10+ | 10+ |
| Git | 2.30+ | 2.40+ |

Установка из архива

1. Распаковка

```
tar -xzf taskitnow-distro-X.Y.Z.tar.gz
cd taskitnow
```

2. Загрузка Docker-образов

```
./install.sh
```

Скрипт проверит наличие Docker, контрольные суммы образов и загрузит все образы в Docker.

3. Настройка конфигурации

```
cp .env.example .env
nano .env
```

Обязательные параметры для изменения:

- POSTGRES_PASSWORD — пароль PostgreSQL
- RMQ_PASS — пароль RabbitMQ (обновить также в RMQ_URL)
- MONGO_PASS — пароль MongoDB
- ACCESS_SECRET, REFRESH_SECRET — секреты JWT-токенов (минимум 32 символа)
- DOMAIN, API_DOMAIN — домены приложения

4. Настройка DNS / hosts

Добавьте записи в /etc/hosts (или настройте DNS-сервер):

```
127.0.0.1 taskitnow.local
127.0.0.1 api.taskitnow.local
```

5. Выбор сервисов (опционально)

Файл services.conf определяет, какие сервисы будут запущены. По умолчанию включены все основные сервисы. Закомментируйте строки для отключения:

```
# Отключить сервис — добавить # в начало строки
##t-tg-bot
##t-ews
```

6. Запуск

```
./start.sh
```

7. Проверка

```
./status.sh
```

После запуска:

- Фронтенд: <http://taskitnow.local>
- API: <http://api.taskitnow.local>
- RabbitMQ Management: <http://localhost:15672>

Управление

| Команда | Описание |
|------------|---------------------|
| ./start.sh | Запуск всей системы |

| Команда | Описание |
|----------------------------------|------------------------------------|
| <code>./stop.sh</code> | Остановка всех контейнеров |
| <code>./stop.sh --volumes</code> | Остановка + удаление данных (тома) |
| <code>./status.sh</code> | Статус всех сервисов |

Конфигурация

.env

Основной файл конфигурации. Содержит параметры для всех компонентов:

- Домены и протокол
- Учётные данные баз данных (PostgreSQL, MongoDB)
- Параметры RabbitMQ
- JWT-секреты
- URL сервисов для фронтенда (NEXT_PUBLIC_SERVICE_*)

services.conf

Текстовый файл со списком включённых сервисов. Закомментированные строки (начинающиеся с #) — отключённые сервисы.

nginx

Шаблон маршрутизации `nginx/taskitnow.conf.template` автоматически заполняется при запуске значениями `DOMAIN` и `API_DOMAIN` из `.env`.

SSL-сертификаты

По умолчанию Nginx слушает на порту 80 (HTTP). Для HTTPS:

Самоподписанный сертификат

```
mkdir -p ssl
openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout ssl/privkey.pem \
  -out ssl/fullchain.pem \
  -subj "/CN=taskitnow.local"
```

Затем добавьте в `nginx/taskitnow.conf.template` блок для HTTPS-сервера и перезапустите:

```
./stop.sh && ./start.sh
```

Устранение неполадок

Сервис не запускается

Проверьте логи конкретного сервиса:

```
docker compose -f docker-compose.services.yml logs tt-auth
```

Ошибки подключения к БД

Убедитесь, что PostgreSQL создал все базы:

```
docker compose -f docker-compose.infra.yml exec postgresql psql -U postgres -c "\l"
```

RabbitMQ не готов

Проверьте статус:

```
docker compose -f docker-compose.infra.yml exec rabbitmq rabbitmq-diagnostics -q status
```

Пересоздание баз данных

```
./stop.sh --volumes
```

```
./start.sh
```