

**Руководство администратора
Универсальной интеграционной
платформы БОЛТ**

Москва, 2026

Оглавление

1. Общие сведения.....	3
2. Требования к инфраструктуре.....	3
3. Установка и первичный запуск.....	3
3.1. Передача дистрибутива на сервер.....	3
3.2. Установка Docker (если не установлен).....	4
3.3. Установка Платформы.....	4
3.4. Первый вход.....	4
4. Управление пользователями и правами.....	5
4.1. Роли пользователей.....	5
4.2. Основные операции.....	5
5. Ведение каталога продуктов.....	5
6. Настройка форм и обработки заявок.....	5
6.1. Подключение и сопровождение API-режима.....	6
7. Резервное копирование и восстановление.....	6
8. Обновление Платформы.....	7
9. Сопровождение и разрешение проблем.....	7

1. Общие сведения

Настоящее руководство предназначено для администратора программного обеспечения «Универсальная интеграционная платформа БОЛТ» (далее — Платформа) и описывает установку, настройку, сопровождение и администрирование экземпляра системы.

Платформа — прикладная информационная система класса веб-витрины заказов программного обеспечения, работающая как веб-ориентированное решение с пользовательским интерфейсом, серверной частью, базой данных и административными средствами настройки.

Администратор отвечает за работоспособность экземпляра, ведение каталога продуктов, управление учётными записями и правами, настройку форм и обработки заявок, резервное копирование и обновление.

При подключении API-режима администратор взаимодействует со специалистами компании-разработчика, которые по отдельному запросу партнера настраивают автоматизированный прием заявок и доступ к нему.

2. Требования к инфраструктуре

Параметр	Требование
Операционная система	Ubuntu Server 24.04 LTS (рекомендуемая и проверенная конфигурация)
CPU	8 и более ядер
RAM	16 ГБ и более
Диск	50 ГБ SSD и более
Docker Engine	версия 24.x и новее
Docker Compose	версия v2.24 и новее
Сетевые порты	13000 (HTTP-интерфейс), 5432 (PostgreSQL)

3. Установка и первичный запуск

3.1. Передача дистрибутива на сервер

Скопировать установочный пакет на сервер с рабочего места администратора:

```
scp megabolt-1.0.0-linux-x86_64.tar.gz user@%IP_адрес_сервера%:/opt/
```

Подключиться к серверу и убедиться, что пакет загружен:

```
ssh user@%IP_адрес_сервера%  
cd /opt  
sudo ls
```

3.2. Установка Docker (если не установлен)

Установите Docker Engine и Docker Compose согласно документации для вашей ОС. Проверить установку:

```
docker --version  
docker compose version
```

3.3. Установка Платформы

На целевом сервере выполнить (подставив свои пути и имена файлов):

```
cd /opt  
sudo tar -xzf megabolt-1.0.0-linux-x86_64.tar.gz  
cd megabolt-1.0.0  
sudo ./install.sh
```

Дождаться окончания установки и загрузки контейнеров. Проверить доступность в браузере:

```
http://%IP_адрес_сервера%:13000/
```

3.4. Первый вход

Авторизация выполняется с учётной записью администратора по умолчанию:

Параметр	Значение по умолчанию
Логин	admin@nocobase.com
Пароль	Ty1_wNqK8f

ВАЖНО: сразу после первого входа смените пароль администратора по умолчанию и, при необходимости, адрес учётной записи. Использование стандартных учётных данных в промышленной эксплуатации недопустимо.

4. Управление пользователями и правами

В Платформе используются учётные записи, определяемые логином (адрес электронной почты в нижнем регистре), паролем и назначенной ролью. Разграничение прав выполняется на основе ролевой модели (ACL/RBAC).

Для внешней системы создается отдельная техническая учетная запись. В рассматриваемом профиле используется роль `license_api` с правом создания записей `license_request` без прав просмотра списка, изменения и удаления записей или администрирования. Токен доступа выпускается для этой учетной записи; партнеру также сообщается его идентификатор для поля `partner`.

4.1. Роли пользователей

Роль	Полномочия
Пользователь витрины	Просмотр каталога, оформление и отправка заявок
Оператор	Контроль обработки заявок, сопровождение исполнения операций
Администратор	Настройка, сопровождение, управление доступом и конфигурацией
Техническая учетная запись API	Автоматизированное создание заявок через API в пределах выданных прав; доступ к административным функциям не предоставляется.

4.2. Основные операции

- Создание учётной записи: указать e-mail (логин), задать пароль, назначить роль.
- Изменение роли и прав доступа пользователя.
- Блокировка / отключение учётной записи при выводе пользователя из эксплуатации.
- Сброс пароля пользователя.

5. Ведение каталога продуктов

Администратор поддерживает актуальность перечня программных продуктов, доступных для заказа в экземпляре Платформы.

- Добавление новой позиции продукта в витрину.
- Настройка карточки продукта: наименование, вариант лицензирования, количество устройств, комментарии, дополнительные параметры.
- Редактирование и снятие с публикации позиций каталога.

6. Настройка форм и обработки заявок

- Настройка состава полей формы заявки (получатель, количество лицензий, контактные и идентификационные сведения).

- Настройка сценария обработки заявки: проверка данных, определение доступности продукта, маршрутизация, обращение к источникам лицензий.
- Контроль текущего состояния и статусов операций.

6.1. Подключение и сопровождение API-режима

Подключение выполняется в целях безопасности только специалистами компании-разработчика по отдельному запросу партнера. Самостоятельная регистрация пользователя в веб-витрине не означает предоставления API-доступа. При подключении необходимо:

- Согласовать используемый сценарий, адрес API и идентификатор партнера; настроить техническую учетную запись, ограниченную роль и выдать токен доступа;
- Настроить рабочий процесс для коллекции `license_request`: глобальное событие после действия создания записи, сопоставление параметров и HTTP-запрос в подсистему выдачи лицензий
- Выполнить контрольный запрос с выданными реквизитами, проверить запись заявки, выполнение HTTP-узла и получение тестовой лицензии на контролируемый адрес; отдельно проверить отказ без токена;

Партнер передает токен в `Authorization: Bearer`, роль в `X-Role: license_api` и идентификатор в поле `partner` JSON-заявки. Техническое описание, состав данных и примеры проверок вынесены в отдельное приложение.¹

Успешный ответ метода создания подтверждает регистрацию заявки, но не заменяет проверку ее исполнения. Сбои передачи анализируются в истории выполнений рабочего процесса. Повторную отправку заявки следует согласовывать после проверки результата предыдущего вызова, чтобы не создать дубликат. При этом для тестирования можно без ограничений по попыткам использовать тестовый нетарифицируемый продукт с `serviceid = rktkg_ksos_test`.

При прекращении интеграции либо замене реквизитов специалисты отзывают соответствующий токен или отключают техническую учетную запись. Если рабочий процесс используется несколькими партнерами, его не отключают для блокировки доступа только одного из них.

7. Резервное копирование и восстановление

Для обеспечения надёжности эксплуатации необходимо выполнять регулярное резервное копирование базы данных PostgreSQL и конфигурации Платформы. Рекомендуется настроить резервное копирование по расписанию и контролировать целостность копий.

Следует:

- Создавать резервную копию перед каждым обновлением и изменением конфигурации.
- Хранить резервные копии на отдельном носителе/ресурсе.
- Периодически проверять восстановление копии в тестовом контуре.

¹ См. отдельное приложение «Описание API», версия 1.0 от 23.09.2026 (Описание API.docx), и спецификацию OpenAPI 3.0.3 (bolt-license-api.openapi.yaml).

8. Обновление Платформы

Обновление экземпляра выполняется по утверждённому регламенту:

- Создание резервной копии конфигурации и данных.
- Установка новой версии в тестовом контуре (при наличии).
- Восстановление копии данных в тестовый контур.
- Тестирование основных сценариев (оформление, обработка, выдача, регистрация операции).
- Для подключенного API-режима - проверка токена и роли, приема JSON-заявки, выполнения рабочего процесса и вызова внутреннего сервиса.
- Установка обновления в промышленном контуре и восстановление копии данных.

9. Сопровождение и разрешение проблем

В процессе эксплуатации администратор контролирует работоспособность инфраструктуры, соблюдение регламентов и обрабатывает обращения.

Полезные команды диагностики:

```
# статус контейнеров
sudo docker ps

# просмотр логов приложения
sudo docker logs -f megabolt-app

# перезапуск сервисов
sudo docker restart %имя_контейнера%
```

При наличии действующего контракта на техническую поддержку вендора обращения направляются по адресу электронной почты support@boltcorp.ru с указанием описания проблемы, даты и времени ошибки, сведений о заявке и снимка экрана.

Для API-инцидента дополнительно указываются адрес метода, время вызова, значение partner, идентификатор записи data.id (если получен), HTTP-статус и сообщение об ошибке. При наличии сохраняется x-request-id из ответа. Токены и внутренние ключи из материалов обращения исключаются.

9.1. Идентификация поставочного образа

Приложение поставляется в виде Docker-образа megabolt/app:1.0.0. Версия и состав поставки фиксируются установочным архивом.

9.2. Лицензионная информация компонентов

Состав поставки определяется конкретной версией Docker-образа и перечнем входящих компонентов. Для компонентов указываются версии и применимые лицензионные условия.