АКС Комплекс

Инструкция по установке экземпляра ПО

Программный комплекс автоматизации и контроля процесса подбора и учета продукции «АКС Комплекс»

Версия: 1.0
Дата: 27.10.2026
Правообладатель: ИП Краснов Владимир Андреевич
ИНН: 781422796858 · **ОГРНИП:** 320784700096770
Свидетельство: № 2025687159

1. Общие сведения

Настоящая инструкция описывает установку и запуск проверочного экземпляра программного комплекса автоматизации и контроля процесса подбора и учета продукции «АКС Комплекс» (далее — Комплекс).

Экземпляр предназначен для проверки экспертом функциональных характеристик Комплекса без подключения реального складского оборудования: PTL-модули, информационные дисплеи и световые колонны имитируются встроенными эмуляторами и демо-режимами.

1.1. Состав Комплекса

Компонент	Назначение	Технология
`ascs_api`	Диспетчеризация, маршруты, заказы, очередь наборки, печать, статистика, REST API	Python 3.11, FastAPI, PostgreSQL, Redis
`ascs_ui`	Автоматизированное рабочее место диспетчера	React 19, TypeScript, nginx
`ptl_api`	Управление PTL-модулями по Modbus TCP	Go, Modbus TCP, Redis
`rid_api`	Управление информационными LED-дисплеями по Modbus TCP	Go, Modbus TCP, Redis
`lighthouse_api`	Управление световыми колоннами через HTTP-реле	Go, HTTP CGI, Redis
`display_emulator`	Эмулятор LED-дисплея (3 экземпляра)	Go
`relay_emulator`	Эмулятор HTTP-реле (16 выходов)	Go

1.2. Проверочный экземпляр

Проверочный экземпляр поставляется в виде набора файлов для ручной установки:

- архив `aks-complex-1.0-offline.tar.gz` с Docker-образами;
- `docker-compose.yml`;
- конфигурации сервисов в каталоге `configs/`;
- сценарий наполнения демо-данными в каталоге `seed/`;
- установочный скрипт `install.sh`;
- настоящая инструкция и руководство пользователя.

Экземпляр разворачивается на виртуальной машине или сервере эксперта. Доступ к операционной системе осуществляется под учётной записью, созданной экспертом при установке Ubuntu; предустановленных паролей ОС в комплект не включается. Для входа в АРМ диспетчера предсоздана учётная запись (см. раздел 5).

2. Требования к среде

2.1. Минимальные требования

Параметр	Значение
ОС	Ubuntu 24.04 LTS, x86_64
CPU	4 ядра
RAM	8 ГБ
Диск	20 ГБ свободного места
Docker Engine	24.0 или новее
Docker Compose	v2 (`docker compose`)
Браузер	Chrome, Firefox, Edge, Safari актуальных версий

Допускается использование другой Linux-совместимой ОС при наличии Docker Engine и Docker Compose v2. Для проверочного экземпляра рекомендуется Ubuntu 24.04 LTS.

2.2. Сетевые порты

Порт	Назначение	Доступ
8080	АРМ диспетчера (UI + прокси `/api`)	снаружи
8001	Веб-интерфейс PTL API	снаружи
8002	Веб-интерфейс RID API	снаружи
8010	Веб-интерфейс Lighthouse API	снаружи
8091–8093	Веб-интерфейсы эмуляторов дисплеев	снаружи
8094	Веб-интерфейс эмулятора реле	снаружи
5432	PostgreSQL	только внутри Docker-сети
6379	Redis	только внутри Docker-сети

Если порт 8010 занят, его можно переопределить переменной `LIGHTHOUSE_PORT` при запуске.

3. Установка offline-экземпляра

3.1. Подготовка

1. Установите Docker Engine и Docker Compose v2 согласно официальной документации Docker.
2. Проверьте установку:

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

3. Распакуйте архив:

```
tar -xzf aks-complex-1.0-offline.tar.gz
cd aks-complex-1.0
```

3.2. Запуск

```
sudo ./install.sh --offline
```

Скрипт выполняет следующие действия:

1. загружает Docker-образы из каталога `images/`;
2. запускает контейнеры без сборки;
3. ожидает готовности API;
4. наполняет базу демонстрационными данными;
5. снимает паузу, которая устанавливается при старте backend;
6. выводит адреса и учётные данные.

Ожидаемый результат в конце установки:

```
АКС Комплекс запущен.
АРМ диспетчера: http://localhost:8080/
...
Вход в АРМ:
  роль:     Диспетчер
  пароль:   AksDemo2026
```

3.3. Установка из исходников

Для сборки из исходников используется тот же скрипт без флага `--offline`:

```
sudo ./install.sh
```

В этом случае Docker Compose собирает образы из соседних каталогов `ascs_api`, `ascs_ui`, `ptl_api`, `rid_api`, `lighthouse_api`, `display_emulator`, `relay_emulator`. Данный режим предназначен для разработки и не требуется эксперту при наличии offline-архива.

4. Проверка работоспособности

4.1. Проверка API

```
curl http://localhost:8080/api/_health
```

Ожидаемый ответ:

```
{"postgres": true, "redis": true, "ptl": true, "rid": true}
```

4.2. Проверка веб-интерфейсов

Адрес	Что должно открыться
`http://<host>:8080/`	Экран входа в АРМ «АКС Комплекс»
`http://<host>:8001/`	Веб-интерфейс PTL API, вкладка «Обзор»
`http://<host>:8002/`	Веб-интерфейс RID API, вкладка «Обзор»
`http://<host>:8010/`	Веб-интерфейс Lighthouse API, вкладка «Обзор»
`http://<host>:8091/`	Эмулятор дисплея № 1
`http://<host>:8094/`	Эмулятор реле

4.3. Проверка состояния оборудования

1. Войдите в АРМ под ролью «Диспетчер».
2. Откройте «Управление» → «Состояние оборудования».
3. Убедитесь, что контроллеры PTL, дисплеи и реле отображаются как онлайн.
4. Откройте «Состояние индикаторов» — должны отображаться 26 PTL-модулей.

5. Учётные данные

Роль	Логин	Пароль	Права
Диспетчер	выбор роли «Диспетчер»	`AksDemo2026`	полный доступ
Оператор	выбор роли «Оператор»	не требуется	чтение и тестирование оборудования

Пароль диспетчера хранится в базе в виде bcrypt-хэша. Для смены пароля:

```
docker compose exec ascs-api python manage.py set-password 'НовыйПароль'
```

6. Демонстрационный сценарий

В базу заранее загружены:

- 3 маршрута: `ДЕМО-1`, `ДЕМО-2`, `ДЕМО-3`;
- 10 заказов (`1001` – `1010`);
- 26 PTL-модулей на трёх контроллерах;
- 3 информационных дисплея;

- 5 световых колонн;
- 3 сотрудника, 2 АРМа, зоны и секции.

6.1. Запуск маршрута

1. Войдите в АРМ как диспетчер.
2. На экране «Очередь комплектации» выберите маршрут `ДЕМО-1`.
3. Нажмите «Пуск» (или откройте маршрут и нажмите «Запустить»).
4. Убедитесь, что заказ `1001` перешёл в статус «Комплектация», а на PTL-модулях 1–3 контроллера `10.0.0.1` загорелся синий цвет.
5. Откройте «Управление» → «Состояние индикаторов» — активные PTL должны быть отмечены как «Активен», цвет «Синий».

6.2. Работа с заказом

1. Откройте заказ `1001` из маршрута.
2. Проверьте позиции, ячейки, количество ящиков и штук.
3. Введите фактическое количество подобранного, если оно отличается от планового.
4. Подтвердите тару и опт (кнопки «Стеллаж» / «Опт»).
5. Завершите заказ. Система перейдёт к следующему заказу маршрута.

6.3. Дисплеи и световые колонны

При активной секции Комплекс отправляет текст на информационные дисплеи и включает световые колонны. Проверить результат можно:

- в веб-интерфейсе RID API (`:8002`) — вкладка «Обзор»;
- в веб-интерфейсе Lighthouse API (`:8010`) — вкладка «Обзор»;
- в эмуляторах дисплеев (`:8091` – `:8093`) и реле (`:8094`).

6.4. Пауза, сброс и повторный запуск

- «На паузу» — приостанавливает маршрут с сохранением состояния;
- «Запустить» — возобновляет маршрут;
- «Сбросить» — возвращает заказы в исходное состояние и гасит оборудование;
- «Полный сброс комплектации» — доступен в разделе «Управление» и сбрасывает все маршруты.

7. Остановка, обновление и удаление

7.1. Остановка

```
./install.sh --down
```

7.2. Просмотр логов

```
./install.sh --logs
```

Или отдельно:

```
docker compose logs -f ascs-api
docker compose logs -f ptl-api
```

7.3. Обновление

1. Остановите Комплекс.
2. Распакуйте новый архив в отдельный каталог.
3. Выполните `sudo ./install.sh --offline`.

Конфигурации и данные хранятся в Docker-томах `aks-complex_aks_pgdata` и `aks-complex_aks_logs` и сохраняются при обновлении. Перед обновлением рекомендуется сделать резервную копию базы:

```
docker compose exec -T db pg_dump -U aks aks > backup_$(date +%F).sql
```

7.4. Полное удаление

```
./install.sh --reset
```

Команда удаляет контейнеры и тома с данными. Действие необратимо.

8. Диагностика типовых проблем

Симптом	Причина	Решение
`address already in use`	занят порт 8010 или 8080	остановить конфликтующий сервис или задать `LIGHTHOUSE_PORT`
`ascs-api is unhealthy`	ошибка миграции или нехватка памяти	`docker compose logs ascs-api`, проверить RAM и диск
PTL/RID/Lighthouse offline	контейнер не запущен или нет Redis	`docker compose ps`, `docker compose logs <service>`
Не удаётся войти	неверный пароль	`docker compose exec ascs-api python manage.py set-password '...'`
Демо-данные не появились	импорт CSV не завершился	`docker compose exec ascs-api python /seed/seed_demo.py`

Симптом	Причина	Решение
Экран «Наборка поставлена на паузу»	backend перезапускался	нажать «Понял» или `curl -X POST http://localhost:8080/api/system/paused_state/ack`

9. Контакты технической поддержки

Техническая поддержка	support@whsoft.ru
Общие вопросы	info@whsoft.ru
Сайт	https://whsoft.ru/
Документация	https://whsoft.ru/docs/ascs/
Правообладатель	ИП Краснов Владимир Андреевич, ИНН 781422796858

Для консультации по развёртыванию и настройке экземпляра рекомендуется обращаться на `support@whsoft.ru` с темой «Проверочный экземпляр АКС Комплекс».