

Руководство по разворачиванию программного обеспечения

АСКТ-Компас

в среде Docker на ОС Astra Linux

1. Программно Аппаратные требования

Требования программному окружению

- ОС Astra Linux
- Docker 25.0.5.astra3+
- Docker Compose 2.40+

Аппаратные требования для работы ПО при условии подключения до 300 контролеров, с периодом хранения данных 180 дней:

- процессор Intel Core i3-12100F
- оперативна память 16 ГБ
- видеокарта встроенная в процессор
- объем жесткого диска 512 ГБ
- операций ввода вывода для жестких дисков 10000 IOPS
- применение RAID контролера с резервным питанием

2. Используемые компоненты и языки

Список внутренних сторонних компонентов «АСКТ-Компас», необходимых для функционирования ПО и/или используемых в его составе:

Сторонний компонент/сервис	Название лицензии	Ссылка на репозиторий
MySQL Community Edition	GPLv2	https://dev.mysql.com/downloads/mysql/?os=src
FrameWork Dotnet 8	NET8.0	https://github.com/dotnet/core/blob/main/release-notes/8.0/8.0.4/8.0.4.md
AspNetCore 8	AspNetCore 8	https://github.com/TekSoftOne/aspnetcore-8
Docker	Apache 2.0	https://github.com/docker
React v16.14	MIT License	https://github.com/facebook/react/releases

3. Установка ПО

3.1 Установка ПО с использованием файла скриптов

Входим в операционную систему, запускаем терминал, далее все действия по настройке будут производиться в нем.

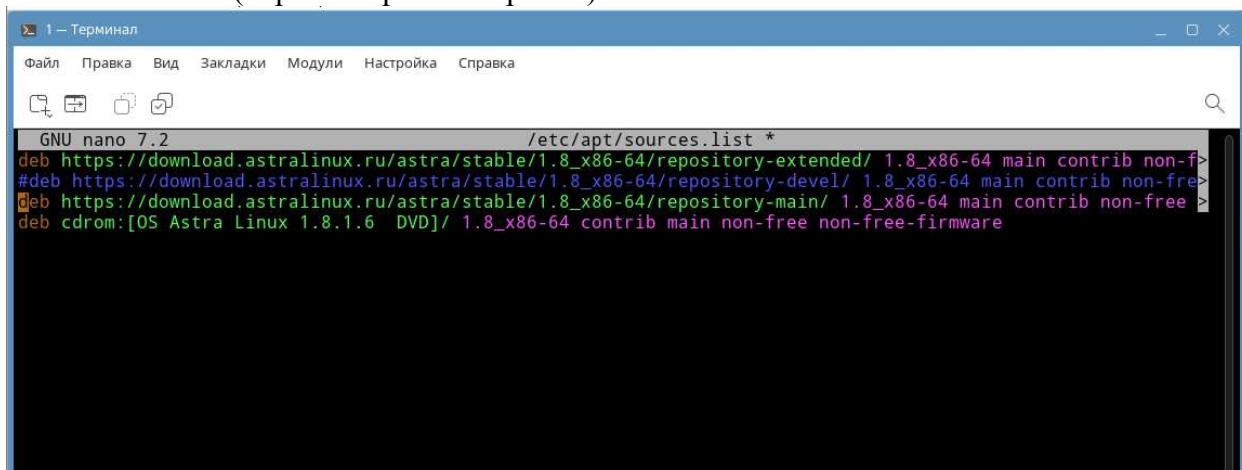
3.1.1 Подключение базовых репозитариев

В текстовом редакторе папо снимем комментарии с репозитариев для этого нужно выполнить команды.

sudo su

Вводим пароль пользователя root **nano /etc/apt/sources.list**

убрать символ «#» перед репозиториями *repository-main* и *repository-extended* (первая и третья строчки)



```
GNU nano 7.2 /etc/apt/sources.list *
deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-extended/ 1.8_x86-64 main contrib non-free
#deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-devel/ 1.8_x86-64 main contrib non-free
deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-main/ 1.8_x86-64 main contrib non-free
deb cdrom:[OS Astra Linux 1.8.1.6 DVD]/ 1.8_x86-64 contrib main non-free non-free-firmware
```

Для сохранения необходимо нажать комбинацию клавиш CTRL+X, потом Enter и соглашаемся с перезаписью Y и еще раз нажимаем Enter

3.1.2 Получение файла скрипта

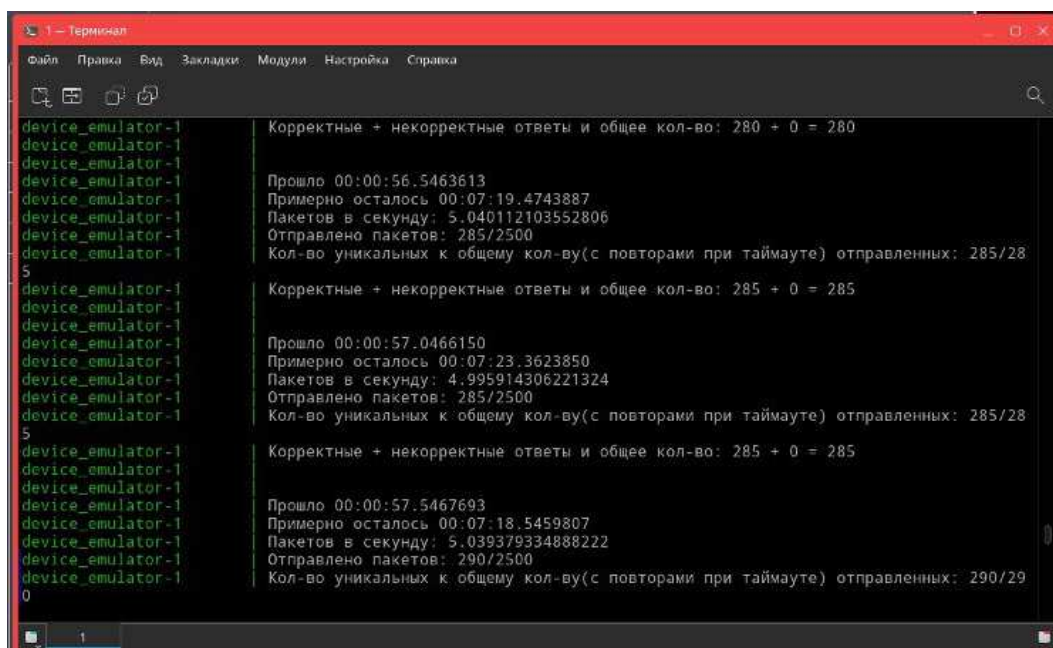
curl -O https://пелизы.блуктех.рф/Releases/install_docker.sh

3.1.3 Запуск файла скрипта

bash /home/radmin/install_docker.sh

3.1.4 Переход на работу с приложением

Дожидаемся окончания работы скрипта, когда на экране терминала будут идти повторяющиеся записи, см рисунок ниже.



```
device_emulator-1 Корректные + некорректные ответы и общее кол-во: 280 + 0 = 280
device_emulator-1
device_emulator-1 Прошло 00:00:56.5463613
device_emulator-1 Примерно осталось 00:07:19.4743887
device_emulator-1 Пакетов в секунду: 5.040112103552806
device_emulator-1 Отправлено пакетов: 285/2500
device_emulator-1 Кол-во уникальных к общему кол-ву(с повторами при таймауте) отправленных: 285/285
5
device_emulator-1 Корректные + некорректные ответы и общее кол-во: 285 + 0 = 285
device_emulator-1
device_emulator-1 Прошло 00:00:57.0466150
device_emulator-1 Примерно осталось 00:07:23.3623850
device_emulator-1 Пакетов в секунду: 4.995914306221324
device_emulator-1 Отправлено пакетов: 285/2500
device_emulator-1 Кол-во уникальных к общему кол-ву(с повторами при таймауте) отправленных: 285/285
5
device_emulator-1 Корректные + некорректные ответы и общее кол-во: 285 + 0 = 285
device_emulator-1
device_emulator-1 Прошло 00:00:57.5467693
device_emulator-1 Примерно осталось 00:07:18.5459807
device_emulator-1 Пакетов в секунду: 5.039379334888222
device_emulator-1 Отправлено пакетов: 290/2500
device_emulator-1 Кол-во уникальных к общему кол-ву(с повторами при таймауте) отправленных: 290/290
0
```

Переходим к пункту 4.2 инструкции для проверки запуска графического интерфейса пользователя.

!!! ВНИМАНИЕ. После работы скрипта контейнеры остаются в запущенном состоянии, и в пункте 4 нужно пропустить подпункты 4.1.1 и 4.1.2 и сразу переходить к запуску графического интерфейса пользователя ПО пункт 4.2.

3.2 Установка Docker и Docker Compose в ручном режиме

Входим в операционную систему, запускаем терминал, далее все действия по настройке будут производиться в нем.

3.2.1 Подключение базовых репозитариев

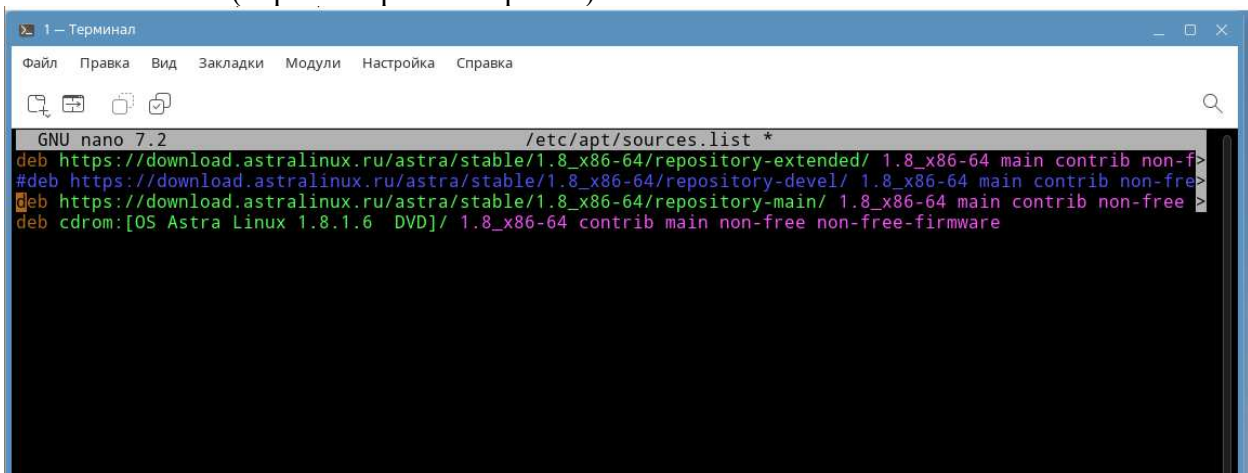
В текстовом редакторе nano снимем комментарии с репозитариев для этого нужно выполнить команды.

`sudo su`

Вводим пароль пользователя root

`nano /etc/apt/sources.list`

убрать символ «#» перед репозитариями *repository-main* и *repository-extended* (первая и третья строки)



```
GNU nano 7.2 /etc/apt/sources.list *
deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-extended/ 1.8_x86-64 main contrib non-free
deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-devel/ 1.8_x86-64 main contrib non-free
deb https://download.astralinux.ru/astra/stable/1.8_x86-64/repository-main/ 1.8_x86-64 main contrib non-free
deb cdrom:[OS Astra Linux 1.8.1.6 DVD]/ 1.8_x86-64 contrib main non-free non-free-firmware
```

Для сохранения необходимо нажать комбинацию клавиш CTRL+X, потом Enter и соглашаемся с перезаписью Y и еще раз нажимаем Enter

3.2.2 Обновление ОС

Установим дополнения операционной системы для этого введем команду

`apt update`

3.2.3 Установка утилиты Среды виртуализации Docker

`apt-get install docker.io`

3.2.4 Создаем папку для загрузки плагинов Docker

`sudo mkdir -p /usr/local/lib/docker/cli-plugins`

3.2.5 Загружаем Docker Compose

`sudo curl -L`

`"https://github.com/docker/compose/releases/latest/download/doc`

**`ker-compose-$(uname -s)-$(uname -m)" \ -o
/usr/local/lib/docker/cli-plugins/docker-compose`**
!!! Внимание перед командой -o должно стоять два пробела, все пишется в одну строчку

3.2.6 Выдаем права на папку

`chmod +x /usr/local/lib/docker/cli-plugins/docker-compose`

3.2.7 Для того, чтобы убедиться в нормальной установке приложения Docker Compose, вводим команду для получения номера версии приложения и убеждаемся, что на этот запрос пришел ответ в виде версии установленного приложения

`docker compose version`

3.2.8 Выходим из режима супер пользователя

`exit`

3.3 Установка ПО АСКТ-Компас

3.3.1 Создать папку в директории home пользователя для установки в нее контейнеры с ПО АСКТ Компас.

`sudo mkdir -p /home/<username>/asktdocker`

Вариант установки с получением файла по сети

3.3.2 Получить файл с ПО по сети интернет

`curl -O https://пелизы.бликтех.рф/Releases/askt_kompas_docker.zip`

3.3.3 Разархивировать файл в папку asktdocker

`unzip askt_kompas_docker.zip -d asktdocker`

! Перейти к пункту 3.2.10 для развертывания контейнеров

Вариант установки с записью файла на Flash накопитель

3.3.4 Скачать архив с контейнерами программного обеспечения, по ссылке

`https://пелизы.бликтех.рф/Releases/askt_kompas_docker.zip`

Разархивировать его на Flash накопитель, на накопителе должна создаваться папка docker

3.3.5 Вставить Flash накопитель в компьютер на который будет производиться установка ПО и примонтировать Flash накопитель

`sudo mkdir -p /mnt/z`

3.3.6 Найти нужный Flash накопитель в списке подключенных устройств

`lsblk`

3.3.7. Примонтировать Flash накопитель

`sudo mount /dev/sdb1 /mnt/z`

3.3.8 Скопировать папку docker в директорию **/home/<username>/asktdocker**

`sudo cp -r /mnt/z/docker/ /home/<username>/asktdocker`

3.3.9 Отмонтировать Flash накопитель

`sudo umount /mnt/z`

Сборки и запуск контейнеров

3.3.10 Перейти в папку с Docker контейнерами проверить что в указанной папки присутствует файл docker-compose.yml для этого выполнить команды

`cd /home/<username>/asktdocker`
`ls`

3.3.11 Запустить восстановление контейнеров и сразу запустить их на исполнение

`sudo docker compose up --build`

После запуска произойдет однократное скачивание необходимого программного окружения и запустятся контейнеры и можно будет перейти к работе с приложением.

4. Запуск и использование ПО

4.1 Запуск ПО из контейнеров

4.1.1 Перейти в папку с Docker контейнерами проверить что в указанной папке присутствует файл docker-compose.yml

`cd /home/<username>/asktdocker`

4.1.2 Запустить контейнеры на исполнение

`sudo docker compose up -d`

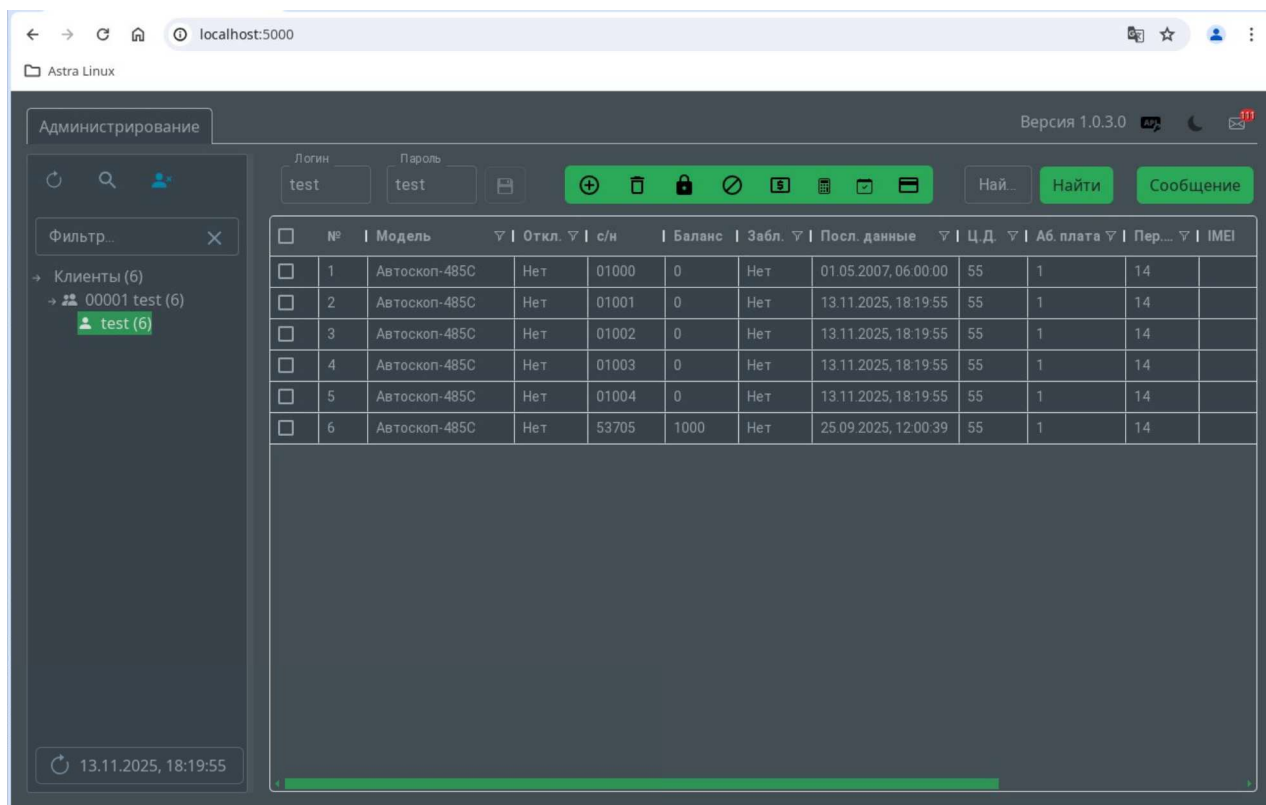
4.2 Использование ПО

4.2.1 Запуск графического интерфейса пользователя

Для запуска графического интерфейса пользователя нужно запустить браузер Chromium и ввести в адресную строку

`localhost:5000`

Графический интерфейс пользователя изображен на рисунке ниже.



4.2.2 Подробное описание использования ПО

Более подробное описание работы с ПО АСКТ-Компас можно посмотреть в разделе 4 Руководства по эксплуатации доступной по ссылке

<https://релизы.бликтех.рф/Releases/РЭ АСКТ-КОМПАС ред3.pdf>

4.2.3 Проверка работоспособности модуля ModelDevice при помощи эмулятора данных

Первые 10 минут работы приложения запущен контейнер с эмулятором пакетов отсылаемыми бортовыми контролерами на сервер. Для проверки, что данные поступают нужно перейти в дереве клиентов в пользователя «test» и при нажатии кнопки обновления экрана которая расположена под деревом клиентов у контролеров с номерами 1001, 1002, 1003, 1004 будут изменяться показатели времени последних полученных данных. На рисунке ниже кнопка обозначенная цифрой один служит для обновления данных, область обозначенная цифрой два служит для отображения времени получения последнего данного.

4.2.4 Проверка работоспособности API

API интерфейс работает на порту 5001 в протоколе SOAP. API позволяет получать список контролеров назначенный на определенный логин, а так же получать сущность трэк, которая содержит параметры движения автомобиля за указанный промежуток времени.

Для проверки работы API интерфейса в терминале можно ввести команду

`curl http://localhost:5001/Soap.asmx`

При этом в терминал будет получен ответ от SOAP сервера в XML формате по доступным методам.

Для проверки работы методов можно запросить версию программного обеспечения введя команду:

```
curl -X POST -H "Content-Type: application/soap+xml; charset=utf-8" -d '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cen="http://center.soap.org/"> <soapenv:Header/> <soapenv:Body> <cen:GetAppVersion/> </soapenv:Body> </soapenv:Envelope>' http://localhost:5001/Soap.asmx
```

На что будет получен ответ от сервера:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <s:Body>
    <GetAppVersionResponse xmlns="http://center.soap.org/">
      <GetAppVersionResult xmlns:d4p1="http://schemas.datacontract.org/2004/07/Center.Models.Soap" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4p1:ErrorFlag>false</d4p1:ErrorFlag>
        <d4p1:ErrorText i:nil="true" />
        <d4p1:Version>1.0.3.0</d4p1:Version>
      </GetAppVersionResult>
    </GetAppVersionResponse>
  </s:Body>
```

Где параметр «d4p1:Version» будет как раз отображать версию программного обеспечения и будет равен 1.0.3.0

Для комфортной работы с API рекомендуем установить утилиту SoapUI, создать в ней новый проект и подключиться удаленным компьютером к ПО АСКТ Компас. Адрес сервера можно узнать набрав в терминале команду

ip -4 addr show scope global

После чего запустить SoapUI создать новый проект и ввести адрес сервера пример см. на рисунке ниже:

Project Name: на свое усмотрение

Initial WSDL: <http://адрес сервера:5001/Soap.asmx>

New SOAP Project

New SOAP Project
Creates a WSDL/SOAP based Project in this workspace

Project Name: Soap


Initial WSDL: http://192.168.1.117:5001/Soap.asmx Browse...

Create Requests: ☒ Create sample requests for all operations?

Create TestSuite: ☐ Creates a TestSuite for the imported WSDL

Relative Paths: ☐ Stores all file paths in project relatively to project file (requires save)

OK Cancel

API ключ для работы с API можно посмотреть в окне которое расположено в правом верхнем углу экрана и обозначено иконкой 

Для тестовой базы API ключ имеет значение:
PI8DS07B/LK1A7Be

Для просмотра описания полей методов нужно обратиться в «Руководство по эксплуатации» раздел 6.