

# VisionLabs LUNA CLEMENTINE 2.0

Запуск и активация на ОС Альт

2.18.0

## Содержание

<b>Введение</b>	<b>3</b>
<b>Системные требования</b>	<b>4</b>
Требования к программному обеспечению сервера . . . . .	4
Требования к аппаратному обеспечению сервера . . . . .	4
<b>1 Лицензирование</b>	<b>5</b>
<b>2 Подготовка к запуску</b>	<b>6</b>
2.1 Установка Docker . . . . .	6
2.2 Авторизация в registry . . . . .	7
2.3 Подготовка и распаковка дистрибутива . . . . .	7
2.4 Настройка Сервиса . . . . .	8
2.4.1 Настройка веб-интерфейса . . . . .	15
<b>3 Запуск Сервиса</b>	<b>18</b>
3.1 Добавление аккаунта администратора . . . . .	18

## **Введение**

Настоящий документ описывает процесс установки и настройки сервиса LUNA CLEMENTINE 2.0 версии 2.18.0 (далее — Сервис) на ОС ALT Server 10.2, а также содержит аппаратные требования и программные требования к ПО.

## Системные требования

### Требования к программному обеспечению сервера

Для работы LUNA CLEMENTINE 2.0 необходимо, чтобы выполнялись следующие требования к программному обеспечению.

Ресурс	Рекомендовано
Необходимое ПО	- ПО VisionLabs LUNA PLATFORM 5 версии 5.33.0 или новее (обязательно). - ПО VisionLabs FaceStream 5 версии 5.1.16 или новее (опционально)
Операционная система (ОС)	Любые, поддерживающие Docker

Функциональность LUNA CLEMENTINE 2.0 ограничена, если установлен FaceStream 5 версии 5.1 и старше.

### Требования к аппаратному обеспечению сервера

Для установки LUNA CLEMENTINE 2.0 необходимо, чтобы выполнялись следующие требования к аппаратным ресурсам.

Ресурс	Рекомендовано
Процессор (CPU)	64-битный процессор Intel или AMD, с 4 ядрами и более, с тактовой частотой 2,0 ГГц и выше
Оперативная память (RAM)	8 Гб и выше
Свободное место на диске (HDD/SSD)	20 Гб и выше

## 1. Лицензирование

Для работы сервиса LUNA CLEMENTINE 2.0 требуется [лицензия LUNA PLATFORM 5](#).

## 2. Подготовка к запуску

Данный раздел описывает установку и использование Docker Compose для развертывания Сервиса.

Процесс настройки и установки необходимо выполнять под учетной записью суперпользователя (с root правами).

Ниже приведены команды для ОС Альт. Все описываемые команды необходимо исполнять в терминале.

Установите утилиты:

```
su -  
apt-get install rpm htop unzip curl
```

### 2.1. Установка Docker

Скачайте rpm пакет:

```
wget http://ftp.altlinux.org/pub/distributions/ALTlinux/p10/branch/files/  
noarch/RPMS/docker-compose-1.29.2-alt1.noarch.rpm
```

Установите:

```
rpm -i docker-compose-1.29.2-alt1.noarch.rpm
```

Установите права доступа:

```
chmod +x /usr/local/bin/docker-compose
```

Запустите Docker:

```
systemctl start docker  
systemctl enable docker
```

Проверьте статус Docker:

```
systemctl status docker
```

## 2.2. Авторизация в registry

Для запуска компонентов необходимо получить их образы из `dockerhub.visionlabs.ru` (VisionLabs registry). Перед этим необходима авторизация.

```
docker login dockerhub.visionlabs.ru --username <username> --password <password>
```

Введите логин `<username>` и пароль `<password>`.

Получить их можно, обратившись к специалистам компании VisionLabs.

## 2.3. Подготовка и распаковка дистрибутива

Дистрибутив представляет собой архив вида `clementine_2.18.0`.

Архив не включает зависимости, которые входят в стандартную поставку репозитория ОС Альт и могут быть загружены из открытых источников.

Перед процессом установки поместите файлы дистрибутива в директорию на сервере. Например, в директорию `/root`. В данной директории не должно быть других файлов дистрибутива кроме целевых, используемых для установки конечного продукта.

Выполните следующие действия.

Переключитесь в режим суперпользователя `root`:

```
su -
```

Создайте директорию для распаковки дистрибутива:

```
mkdir -p /opt/clementine
```

Переместите дистрибутив в созданную директорию:

```
mv /root/clementine_2.18.0 /opt/clementine
```

Установите архиватор `unzip`, если он не установлен:

```
apt-get install unzip
```

Перейдите в директорию с дистрибутивом:

```
cd /opt/clementine
```

Распакуйте файлы:

```
unzip clementine_2.18.0.zip
```

Создайте символическую ссылку. Ссылка указывает, что именно текущая версия дистрибутива используется для запуска:

```
ln -s clementine_2.18.0 current
```

## 2.4. Настройка Сервиса

Необходимо настроить Selinux и Firewall, чтобы они не блокировали работу Сервиса.

Настройка Selinux и Firewall в данном руководстве не описывается.

В архиве «clementine\_2.18.0» есть пример развертывания Docker в директории /example-docker (файлы .env и docker-compose.yml).

Настройка Сервиса выполняется в конфигурационном файле .env:

```
### GLOBAL ###

# Where to preserve persistent data

DATA=./
DOCKER_URL=dockerhub.visionlabs.ru/clementine

# username for databases

CLEMENTINE_USERNAME=clementine

# Password for services

CLEMENTINE_PASSWORD=clementine

# CLEMENTINE tags

CLEMENTINE_AUTH_VER=v1.1.8
CLEMENTINE_DEPARTMENTS_VER=v1.1.4
CLEMENTINE_UI_VER=v1.35.0
```



```
VL_NOTIFIER_VER=0.1.15
CLEMENTINE_API_VER=v1.7.0

# 3rd party services tags

POSTGRESQL_VER=12

### Configs ###

# clementine-ui

CLEMENTINE_UI_HOST=clementine_ui
CLEMENTINE_UI_PORT=8080

# clementine-api

CLEMENTINE_API_HOST=clementine_api
CLEMENTINE_API_PORT=9010

# clementine-auth

CLEMENTINE_AUTH_HOST=clementine_auth
CLEMENTINE_AUTH_PORT=9000
CLEMENTINE_AUTH_DOMAIN=127.0.0.1
CLEMENTINE_AUTH_PROTOCOL=HTTP
CLEMENTINE_AUTH_DSN=postgresql://clementine:clementine@clementine_db/
    clementine_auth

# clementine-departments

CLEMENTINE_DEPARTMENTS_HOST=clementine_departments
CLEMENTINE_DEPARTMENTS_PORT=9100
CLEMENTINE_DEPARTMENTS_DSN=postgresql://clementine:clementine@clementine_db/
    clementine_departments

# luna

LUNA_API_HOST=127.0.0.1
LUNA_API_PORT=5000
LUNA_LICENSES_HOST=127.0.0.1
LUNA_LICENSES_PORT=5120
LUNA_ACCOUNT_ID=a0000000-0000-4000-a000-000000000000

# lunastreams
```

```

LUNA_STREAMS_HOST=127.0.0.1
LUNA_STREAMS_PORT=5160

# vl-notifier API

VLNOTIFIER_HOST=vlnotifier
VLNOTIFIER_PORT=9200
VLNOTIFIER_SUBJECT«=Clementine VL »Notifier

# vl-timetracker

VLTIMETRACKER_HOST=127.0.0.1
VLTIMETRACKER_PORT=8003

# vl-access

VLACCESS_HOST=127.0.0.1
VLACCESS_PORT=9091

# OTHER

LOG_LEVEL=1
TELEGRAM_BOT_TOKEN=
SMTP_SERVER=
SMTP_PORT=
SMTP_SENDER=
SMTP_LOGIN=
SMTP_PASSWORD=
    
```

В конфигурационный файл `.env` внесите необходимые настройки, представленные в Таблице 2.

Таблица 2. Параметры конфигурации

Параметр	Описание
<b>Параметры конфигурации БД LUNA CLEMENTINE</b>	
DATA= ./	Точка монтирования данных контейнеров. Указывать необязательно
DOCKER_url=	По умолчанию: <code>dockerhub.visionlabs.ru/clementine</code>
CLEMENTINE_USERNAME=	Название БД

Параметр	Описание
	По умолчанию: clementine
CLEMENTINE_PASSWORD=	Пароль БД
	По умолчанию: clementine
<b>Теги</b>	
	Теги Clementine:
CLEMENTINE_AUTH_VER=	CLEMENTINE_AUTH_VER=v1.1.8;
CLEMENTINE_DEPARTMENTS_VER=	CLEMENTINE_DEPARTMENTS_VER=v1.1.4;
CLEMENTINE_UI_VER=	CLEMENTINE_UI_VER=v1.35.0;
CLEMENTINE_API_VER=	CLEMENTINE_API_VER=v1.7.0
	Теги сторонних компонентов:
VL_NOTIFIER_VER=	VL_NOTIFIER_VER=0.1.15;
POSTGRESQL_VER=	POSTGRESQL_VER=12
<b>Параметры конфигурации CLEMENTINE UI</b>	
CLEMENTINE_UI_HOST=	Адрес сервера CLEMENTINE UI
	По умолчанию: clementine_ui
CLEMENTINE_UI_PORT=	Порт, на котором будет запущен компонент
	По умолчанию: 8080
<b>Параметры конфигурации CLEMENTINE API</b>	
CLEMENTINE_API_HOST=	Адрес сервера CLEMENTINE API
	По умолчанию: clementine_api
CLEMENTINE_API_PORT=	Порт, на котором будет запущен компонент
	По умолчанию: 9010
<b>Параметры конфигурации CLEMENTINE AUTH</b>	
CLEMENTINE_AUTH_HOST=	Адрес сервера CLEMENTINE AUTH
	По умолчанию: clementine_auth
CLEMENTINE_AUTH_PORT=	Порт, на котором будет запущен компонент

Параметр	Описание
CLEMENTINE_AUTH_DOMAIN=	<p>По умолчанию: 9000</p> <p>Домен приложения, используется для генерации ссылок в email. В зависимости от конфигурации сетевого окружения это может быть именной домен, например, clementine.local или IP адрес компонента, доступный в сети. Параметр не является критически важным и необходим только в случае использования адреса электронной почты для регистрации новых пользователей и сброса паролей</p>
CLEMENTINE_AUTH_PROTOCOL=	<p>По умолчанию: 127.0.0.1</p> <p>Протокол, для работы в web-среде. Зависит от текущих настроек инфраструктуры и требований по безопасности</p> <p>Возможные значения:</p> <ul style="list-style-type: none"> <li>- HTTP — протокол без шифрования;</li> <li>- HTTPS — протокол с SSL-шифрованием</li> </ul> <p>По умолчанию: HTTP</p>
CLEMENTINE_AUTH_DSN=	<p>Подключение компонента CLEMENTINE AUTH к БД</p> <p>По умолчанию:            postgresql://clementine:            clementine@clementine_db/            clementine_auth</p>
<b>Параметры конфигурации CLEMENTINE DEPARTMENTS</b>	
CLEMENTINE_DEPARTMENTS_HOST=	Адрес сервера CLEMENTINE DEPARTMENTS

Параметр	Описание
	По умолчанию: clementine_departments
CLEMENTINE_DEPARTMENTS_PORT=	Порт, на котором будет запущен компонент
	По умолчанию: 9100
CLEMENTINE_DEPARTMENTS_DSN=	Подключение CLEMENTINE DEPARTMENTS к БД
	По умолчанию: postgresql://clementine: clementine@clementine_db/ clementine_departments
<b>Параметры конфигурации LUNA PLATFORM 5</b>	
LUNA_API_HOST=	Хост LP5. IP адрес сервера, на котором развернута LP5
	По умолчанию: 127.0.0.1
LUNA_API_PORT=	Порт компонента API LP5
	По умолчанию: 5000
LUNA_LICENSES_HOST=	Хост LUNA LICENSES
	По умолчанию: 127.0.0.1
LUNA_LICENSES_PORT=	Порт компонента LUNA LICENSES.
	По умолчанию: 5120
LUNA_ACCOUNT_ID=	Идентификатор аккаунта пользователя в LP5
	По умолчанию: a0000000 -0000-4000-a000-000000000000
<b>Конфигурация LUNA Streams</b>	
LUNA_STREAMS_HOST=	Адрес сервера LUNA Streams
	По умолчанию: 127.0.0.1
LUNA_STREAMS_PORT=	Порт LUNA Streams
	По умолчанию: 5160
<b>Параметры конфигурации VL NOTIFIER и отправки уведомлений</b>	

Параметр	Описание
VLNOTIFIER_HOST=	Адрес сервера компонента уведомлений По умолчанию: vlnotifier
VLNOTIFIER_PORT=	Порт компонента уведомлений По умолчанию: 9200
VLNOTIFIER_SUBJECT	Текст в заголовке, отправляемого письма с уведомлением По умолчанию: Clementine VL Notifier
<b>Параметры конфигурации модуля учета рабочего времени</b>	
VLTIMETRACKER_HOST=	Адрес сервера компонента учета рабочего времени По умолчанию: 127.0.0.1
VLTIMETRACKER_PORT=	Порт компонента учета рабочего времени По умолчанию: 8003
<b>Параметры конфигурации VL Access</b>	
VLACCESS_HOST=	Адрес сервера VL Access По умолчанию: 127.0.0.1
VLACCESS_PORT=	Порт VL Access По умолчанию: 9091
<b>Параметры конфигурации SMTP</b>	
LOG_LEVEL=	Уровень логирования. Доступные значения: 0 — запись только ошибок; 1 — запись ошибок и информационных сообщений; 2 — запись ошибок, информационных сообщений и отладочной информации По умолчанию: 1

Параметр	Описание
TELEGRAM_BOT_TOKEN=	Токен бота Telegram, куда будут присылаться уведомления
SMTP_SERVER=	Адрес сервера SMTP
SMTP_PORT=	Порт сервера SMTP. Доступные значения: 25 — без шифрования; 465 — SSL; 587 — TLS
SMTP_SENDER=	Email отправителя для отправки уведомлений пользователям. Например, robot@clementine.local
SMTP_LOGIN=	Логин для подключения к SMTP серверу
SMTP_PASSWORD=	Пароль для подключения к SMTP серверу

\*Является примером идентификатора, а не реальным параметром. Необходимо подставить ID из LP5.

Параметры конфигурации почтового сервера SMTP не являются критически важными и необходимы только в случае использования адреса электронной почты для регистрации новых пользователей и сброса паролей.

Параметры конфигурации SMTP также используются для компонента уведомлений.

Ошибки загрузки данных записываются в системный лог посредством подсистемы логирования с тремя уровнями логирования.

Внутренние ошибки записываются в системный лог посредством подсистемы логирования:

- с уровнем важности «ошибка» в случае, если дальнейшее выполнение программы невозможно;
- с уровнем важности «предупреждение», если программа сохраняет работоспособность.

#### 2.4.1. Настройка веб-интерфейса

Настройка веб-интерфейса Сервиса выполняется в конфигурационном файле `config.json` (находится в `example-docker/config/ui/`):

```
{
  "auth": {
    "allowPasswordChange": {
      "ui": true,
      "email": true
    }
  },
  "similarityThresholds": {
    "low": 0.75,
    "medium": 0.95
  },
  "maxImageSize": "2G"
}
```

Таблица 3. Параметры конфигурации веб-интерфейса

Параметр	Описание
allowPasswordChange	<p>Настройка типа регистрации в Сервисе.</p> <p>Всех пользователей в интерфейсе Сервиса создает администратор. При этом новому пользователю приходит ссылка на адрес электронной почты для продолжения регистрации (подробнее см. «VISIONLABS LUNA CLEMENTINE. Руководство пользователя»).</p> <p>Также возможно создание нового пользователя с явным указанием пароля в Сервисе Для этого необходимо настроить возможность задавать пароли в веб-интерфейсе Сервиса.</p> <p>По умолчанию: «ui»: true, «email»: true</p>
similarityThresholds	<p>Настройка цветового обозначения порогов схожести:</p> <ul style="list-style-type: none"> <li>- значение степени схожести ниже «low» будут отмечены красным цветом;</li> <li>- значение степени схожести между «low» и «medium» — желтым цветом;</li> <li>- значение степени схожести выше «medium» — зеленым цветом.</li> </ul> <p>По умолчанию: low: 0.75, medium": 0.95</p>
maxImageSize	<p>Максимальный размер изображения.</p> <p>При изменении этого параметра должен устанавливаться CLIENT_MAX_BODY_SIZE для контейнера clementine-ui в docker-compose.yml.</p>



<b>Параметр</b>	<b>Описание</b>
	По умолчанию: 2G

### 3. Запуск Сервиса

Перейдите в директорию Docker:

```
cd /opt/clementine/current/example-docker
```

Для запуска Сервиса введите команду:

```
./start_clementine.sh
```

Проверьте статус всех запущенных Docker-контейнеров:

```
docker-compose ps
```

#### 3.1. Добавление аккаунта администратора

После того, как Сервис запустится, создайте новую учетную запись и следуйте указаниям:

```
docker-compose exec clementine_auth python3 cli.py createaccount
```

Укажите логин (адрес электронной почты) и пароль учетной записи, установите роль пользователя.

Для последующего входа в Сервис в качестве логина необходимо использовать указанный адрес электронной почты.

Запустите создание новой авторизации для аккаунта и следуйте указаниям:

```
docker-compose exec clementine_auth python3 cli.py createauth
```

Запустите смену роли аккаунта и следуйте указаниям:

```
docker-compose exec clementine_auth python3 cli.py setaccountrole
```