

# **Руководство по развертыванию системы**

## **«АльтерОфис Веб 2026.0.0.0»**

Страниц 53

2025г.

# Оглавление

<b>1. Введение -----</b>	<b>4</b>
1.1. Назначение документа-----	4
1.2. Область применения -----	4
<b>2. Общие сведения о системе -----</b>	<b>5</b>
2.1. Назначение системы -----	5
2.2. Архитектура системы-----	5
2.2.1. Основные уровни системы «АльтерОфис Веб»-----	6
2.2.2. Описание компонентов системы -----	6
2.2.3. Дополнительные компоненты для развертывания системы -----	7
<b>3. Требования -----</b>	<b>10</b>
3.1. Требования к оборудованию-----	10
3.1.1. Вариант 1 -----	10
3.1.2. Вариант 2-----	10
3.1.3. Вариант 3-----	10
3.2. Требования к программному обеспечению-----	10
3.2.1. Требования к серверным операционным системам-----	10
3.2.2. Требования к контейнеризации и оркестрации -----	11
3.2.3. Требования к клиентской части (браузеры) -----	11
3.3. Требования к сетевой инфраструктуре-----	11
3.4. Требования к безопасности -----	11
3.5. Требования к квалификации администратора-----	12
<b>4. Установка и первичная настройка -----</b>	<b>13</b>
4.1. Подготовка к установке-----	13
4.2. Установка -----	16
4.3. Настройка системы для работы с макросами -----	18
4.3.1. Порядок выполнения настроек для работы с макросами -----	18
4.3.2. Подготовка yaml-файлов-----	19
4.3.3. Смена владельца для VOLUME -----	21
4.3.4. Включение макросов -----	21
4.3.5. Применение внесенных изменений -----	22
4.3.6. Загрузка и создание макросов в АльтерОфис Веб -----	23
4.4. Настройка федеративного обмена между серверами -----	24
4.4.1. Порядок выполнения операций для настройки федеративного доступа -----	25
4.4.2. Настройка сетевого взаимодействия -----	26
4.4.3. Проверка сетевой доступности -----	27
4.4.4. Активация приложения Federation -----	28
4.4.5. Настройка доступа для изолированных серверов -----	28
4.4.6. Установка базового URL-адреса -----	29
4.4.7. Настройка «белого» списка IP адресов -----	30
4.4.8. Настройка межсерверного обмена для пользователей -----	30

4.4.9. Настройка доверенных серверов -----	33
4.4.10. Синхронизация адресных книг для федеративного доступа -----	35
4.4.11. Настройка редакторов для совместной работы при федеративном доступе -----	35
<b>5. Резервное копирование и восстановление-----</b>	<b>37</b>
5.1. Резервное копирование АльтерОфис Веб -----	38
5.1.1. Полное копирование системы -----	38
5.1.2. Резервное копирование пользовательских данных -----	40
5.2. Восстановление АльтерОфис Веб из резервных копий -----	42
<b>6. Процедура обновления сертификата и резервное копирование сертификатов-----</b>	<b>45</b>
6.1. Создание резервной копии текущих сертификатов -----	45
6.2. Установка новых сертификатов -----	45
6.3. Восстановление старых сертификатов (в случае ошибки) -----	46
<b>7. Полная деинсталляция -----</b>	<b>48</b>
<b>8. Обновление системы АльтерОфис Веб-----</b>	<b>49</b>
<b>9. Подключение S3-совместимого хранилища как основного хранилища пользовательских данных 50</b>	
9.1. Настройка основного хранилища MinIO на уже развернутой системе-----	50
9.2. Настройка MinIO в качестве основного хранилища при развертывании с нуля -----	51

# 1. Введение

## 1.1. Назначение документа

Настоящее руководство предназначено для системных администраторов и специалистов, осуществляющих развертывание, ввод в эксплуатацию и сопровождение корпоративной системы файлового хостинга и совместной работы на базе решения «АльтерОфис Веб».

Документ описывает архитектуру решения, требования к инфраструктуре, порядок установки и настройки основных компонентов, а также процедуры резервного копирования и восстановления данных.

---

## 1.2. Область применения

Руководство применяется при развертывании системы в инфраструктуре организации с использованием контейнерной оркестрации (Docker Compose), а также может быть использовано для восстановления системы после сбоя или миграции на новое оборудование.

---

## 2. Общие сведения о системе

### 2.1. Назначение системы

Система предназначена для организации защищенного корпоративного файлового хостинга и обмена файлами между пользователями внутри организации.

Система обеспечивает защищённое хранение данных, предоставляет веб-интерфейс для совместной работы и редактирования документов.

Ключевые цели:

- Централизованное хранение и обмен файлами.
- Совместное редактирование офисных документов в реальном времени.
- Обеспечение безопасного доступа к данным как внутри корпоративной сети, так и извне.
- Мониторинг активности и поиск информации по содержимому файлов.

### 2.2. Архитектура системы

АльтерОфис Веб реализован в виде клиент-серверного приложения, взаимодействие с серверной частью осуществляется через браузер.

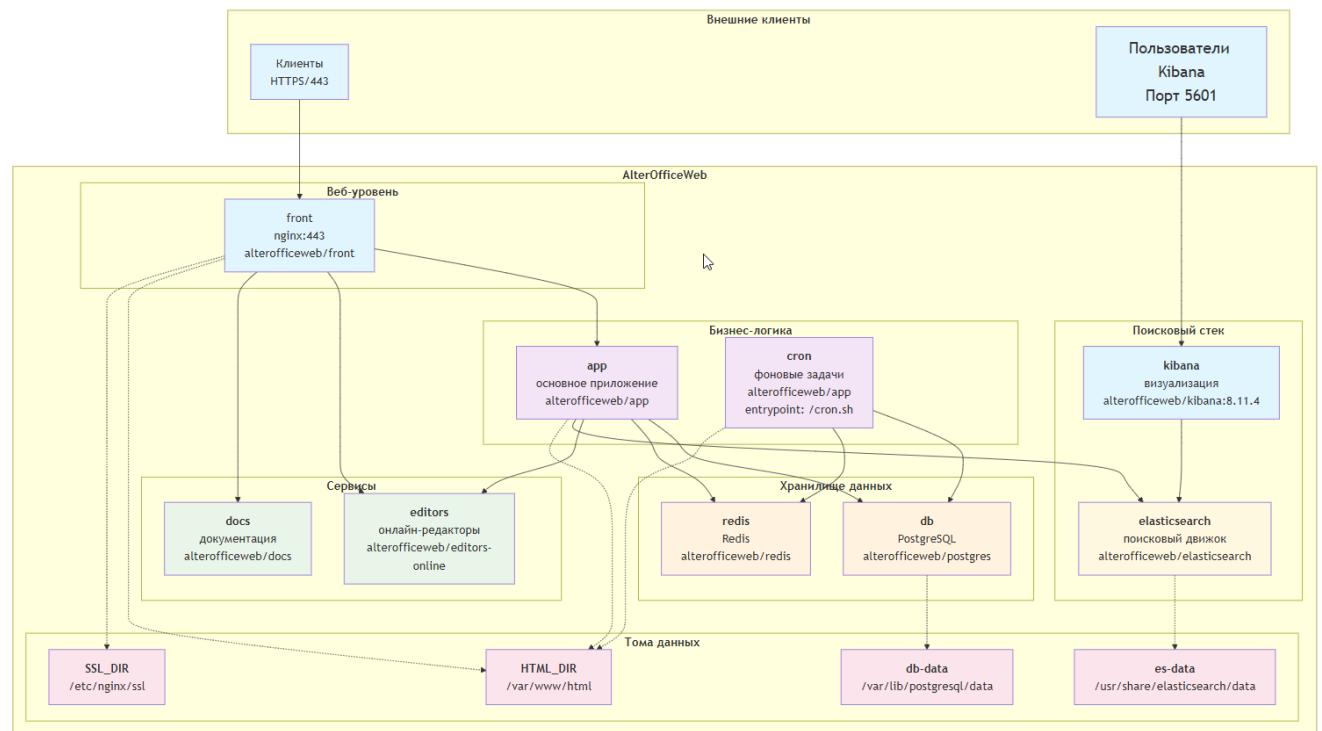


Рисунок 1. Архитектура системы

## 2.2.1. Основные уровни системы «АльтерОфис Веб»

Уровень	Описание
Веб-уровень	Контейнер <b>front</b> , обеспечивает доступ пользователей, перенаправляет запросы в контейнеры <b>app</b> и <b>editors</b> .
Хранилище данных	Контейнеры <b>db</b> и <b>redis</b> .
Бизнес-логика	Контейнеры <b>app</b> (основное приложение) и <b>cron</b> (фоновые задачи).
Сервисы	Контейнеры <b>editors</b> (онлайн-редакторы) и <b>docs</b> (справочная информация по системе).
Поисковый стек	Контейнеры <b>elasticsearch</b> (поисковый движок) и <b>kibana</b> (мониторинг и управление).

## 2.2.2. Описание компонентов системы

Система построена по микросервисной архитектуре, где каждый компонент работает в изолированном Docker-контейнере. Взаимодействие между контейнерами осуществляется через заранее определенные сетевые порты. Внешний веб-трафик поступает на обратный прокси-сервер (Nginx), который маршрутизирует запросы к соответствующим сервисам.

Список контейнеров отражен в таблице ниже.

Уровень / Контейнер	Описание
<b>Веб-уровень</b>	
front	Обеспечивает балансировку нагрузки, кэширование статического содержимого, обеспечивает доступ пользователей, перенаправляет запросы в контейнеры.
<b>Бизнес-логика</b>	
app	Основное веб-приложение, интерпретирует PHP код, реализует серверную логику, обеспечивает хранение файлов, управление пользователями и правами доступа. Взаимодействует с СУБД (PostgreSQL) и кешем (Redis).
cron	Контейнер, обеспечивающий выполнение фоновых задач, таких как обработка файлов, индексация, рассылка уведомлений.
<b>Хранилище данных</b>	
db	Хранение служебной информации, пользователей, метаданных, настроек.
redis	Используется для кэширования запросов к базе данных, блокировки файлов, повышения производительности.
<b>Сервисы</b>	

Уровень / Контейнер	Описание
editors-online	Контейнер обеспечивает совместное редактирование документов (текстовые файлы, таблицы, презентации). Взаимодействует с основным контейнером <b>app</b> по протоколу WOPI.
docs	Контейнер содержит онлайн-документацию.
<b>Поисковый стек</b>	
elasticsearch	Поисковый движок, используется для реализации расширенного полнотекстового поиска по содержимому файлов и метаданным.
kibana	Веб-интерфейс для мониторинга и управления Elasticsearch.

### 2.2.3. Дополнительные компоненты для развертывания системы

При подготовке системы «АльтерОфис Веб» к продуктивному использованию необходимо предусмотреть интеграцию с ключевыми корпоративными сервисами.

Следующие дополнительные компоненты являются важными для промышленной эксплуатации системы и должны быть настроены до ввода в эксплуатацию:

- **SSL/TLS сертификат** обеспечивает безопасность соединения между клиентом и сервером, шифруя передаваемые данные.
- **Внешнее хранилище** данных позволяет расширить объем хранилища данных для документов пользователей.
- **LDAP / AD** позволяет использовать централизованную аутентификацию через корпоративный каталог.
- Подключение системы к **серверу электронной почты** для рассылки уведомлений пользователям.

При планировании использования нескольких экземпляров системы «АльтерОфис Веб» в режиме горизонтального масштабирования (для обслуживания большого числа одновременно работающих пользователей) необходимо предусмотреть вынос пользовательских файлов из локальной файловой системы. Для этого рекомендуется рассмотреть внедрение S3-совместимого объектного хранилища (например, MinIO, Серф и др.) в качестве централизованного хранилища.

Использование объектного хранилища позволит:

- обеспечить корректную работу нескольких экземпляров «АльтерОфис Веб» без необходимости синхронизации локальных каталогов данных;
- снизить нагрузку на файловую систему серверов приложений и улучшить производительность при большом количестве пользователей.

Объектное хранилище должно использоваться исключительно для пользовательских файлов. Метаданные, конфигурация и кэш должны быть вынесены в отдельные, общие для всех узлов сервисы. Логическая архитектура компонентов приведена на рисунке ниже.

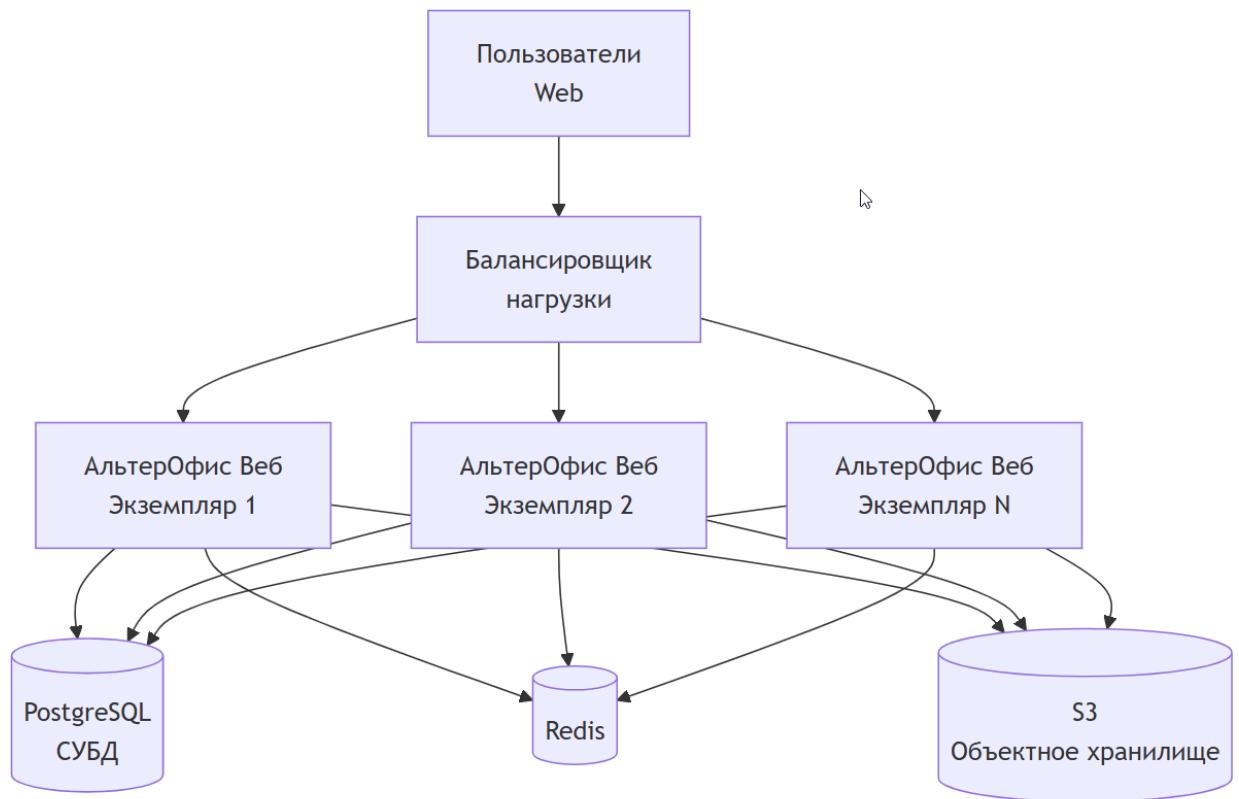


Рисунок 2. Диаграмма компонентов

#### 2.2.4.3. Порты

Для обеспечения корректной работы «АльтерОфис Веб» необходимо, чтобы были открыты и доступны определенные порты.

Перечень используемых портов:

Сервис	Порт	Протокол	Назначение
Веб-сервер (Nginx)	443	HTTPS	Доступ к веб-интерфейсу (шифрованный)
PHP-FPM	9000	TCP	Обработка PHP-запросов (обычно через внутренний прокси Nginx)
PostgreSQL	5432	TCP	Доступ к базе данных
Redis	6379	TCP	Кэширование и очередь задач

Сервис	Порт	Протокол	Назначение
LDAP / Active Directory	389	TCP	Доступ к каталогу пользователей (нешифрованный)
LDAPS / Active Directory	636	TCP	Доступ к каталогу пользователей через SSL
SSO / OAuth2 / SAML	443	HTTPS	Аутентификация через внешние системы
SMTP / STARTTLS	587	TCP	Отправка почты через защищённый канал
SMTP / SSL	465	TCP	Отправка почты через SSL
Cron / Background Jobs	—	—	Локальные задачи (внутренние порты контейнеров)
АльтерОфис Веб Онлайн	9980	HTTP	Интеграция офисного редактора
WebDAV	443	HTTPS	Доступ к файлам через протокол WebDAV
elasticsearch	9200	TCP	REST API поиска
kibana	5601	TCP	Веб-интерфейс мониторинга

### 3. Требования

#### 3.1. Требования к оборудованию

Установка «АльтерОфис Веб» может выполняться как на выделенный сервер, так и в виртуальной среде.

---

##### 3.1.1. Вариант 1

- Общее количество пользователей 100
- Одновременно работающих пользователей 10

Параметр	Минимальные требования
Процессор (CPU)	4 ядра
Оперативная память (RAM)	8 ГБ
Дисковое пространство	80 ГБ (SSD)
Сетевое подключение	100 Мбит/с

---

##### 3.1.2. Вариант 2

- Общее количество пользователей 1000
- Одновременно работающих пользователей 100

Параметр	Минимальные требования
Процессор (CPU)	12 ядер
Оперативная память (RAM)	16 ГБ
Дисковое пространство	240 ГБ (SSD) - зависит от планируемого объема хранимых данных.
Сетевое подключение	1 Гбит/с

---

##### 3.1.3. Вариант 3

Для поддержки более 100 одновременно работающих пользователей, необходима индивидуальная проработка требований. Система позволяет использовать горизонтальное масштабирование и использовать балансировку нагрузки входящих подключений к серверам.

---

#### 3.2. Требования к программному обеспечению

##### 3.2.1. Требования к серверным операционным системам

Рекомендованные версии серверных операционных систем:

- АльтерОС 2025 (9.6)
- Astra Linux 1.8.2
- РЕД ОС 8.0
- Ubuntu 24.04

---

### 3.2.2. Требования к контейнеризации и оркестрации

Рекомендованные версии для серверных операционных систем:

Серверная ОС	Рекомендованная версия Docker	Рекомендованная версия Docker Compose
АльтерОС 2025 (9.6)	20.10.25	2.26.1
Astra Linux 1.8.2	25.0.5	1.29.0
РЕД ОС 8.0	28.1.1	2.39.1
Ubuntu 24.04	28.5.1	2.40.2

---

### 3.2.3. Требования к клиентской части (браузеры)

Рекомендованные версии браузеров:

- Яндекс Браузер. Версия: 25.8.5.983 (64-разрядный)
- Google Chrome. Версия: 141.0.7390.123
- Mozilla Firefox. Версия: 144.0 (64-разрядный)

---

## 3.3. Требования к сетевой инфраструктуре

- Статический IP-адрес для сервера.
- Зарегистрированное доменное имя (FQDN), направленное на IP-адрес сервера.
- Валидный SSL-сертификат для доменного имени.
- Открытые и перенаправленные порты 80 и 443 на хост-систему.
- Стабильное интернет-соединение с достаточной исходящей скоростью для передачи файлов пользователям.
- Поддержка DNS для доступа к контейнерам по именам.
- Ограничение внешнего доступа к внутренним сетям Docker.

---

## 3.4. Требования к безопасности

- Запрет доступа по SSH с использованием пароля, использование ключей.
- Регулярное обновление системных пакетов и образов Docker.
- Использование сложных паролей для всех учетных записей (СУБД, администратора системы).

- Изоляция контейнеров в отдельной сети Docker.
- Использование TLS-сертификатов для всех внешних соединений.

---

### 3.5. Требования к квалификации администратора

Администратор, выполняющий развертывание и сопровождение системы, должен обладать знаниями и навыками в областях:

- Основы администрирования Linux-систем.
- Основы работы с Docker и Docker Compose.
- Основы работы с базами данных PostgreSQL.
- Базовое понимание сетевых технологий (DNS, TCP/IP, настройка брандмауэра).
- Настройка и обслуживание веб-сервера Nginx.
- Настройка TLS-сертификатов
- Умение работать с системными логами.

---

## 4. Установка и первичная настройка

Развертывание системы «АльтерОфис Веб» будет рассмотрено на примере:

- **Экземпляр 1** будет иметь адрес <https://demo03-web2025.alteroffice.ru>

Параметры, которые потребуются для установки:

Параметр	Описание	Пример
<b>DEPLOYMENT_PATH_SERVER</b>	Путь к директории развертывания на хосте	/opt/alterofficeweb/
<b>PATH_SSL</b>	Путь к директории с SSL сертификатами	/etc/ssl/alteroffice
<b>SERVER_IP</b>	IP-адрес хоста сервера, где установлен контейнер <b>app</b> для <b>Экземпляр 1</b>	172.20.18.10
<b>SERVER_URL</b>	URL доступа к системе АльтерОфис Веб, <b>Экземпляр 1</b>	<a href="https://demo03-web2025.alteroffice.ru">https://demo03-web2025.alteroffice.ru</a>
<b>SERVER_HOSTNAME</b>	Доменное имя или FQDN (Fully Qualified Domain Name) сервера АльтерОфис Веб.	demo03-web2025.alteroffice.ru
<b>SERVICE_USER_SERVER</b>	Пользователь для развертывания и управления контейнерами <b>Экземпляр 1</b>	gitlab-runner

### 4.1. Подготовка к установке

Для установки «АльтерОфис Веб» подготовьте выделенный или виртуальный сервер с требуемыми характеристиками.

**Шаг 1.** Убедитесь, что **сетевые порты 443 и 5601 правильно настроены**: правила фаервола разрешают входящий и исходящий трафик на эти порты для подготовленного сервера.

**Шаг 2.** Убедитесь, что **есть валидные SSL сертификаты** с именами **server.key** и **server.crt**.

**Шаг 3.** Установите Docker и Docker Compose.

**Шаг 4.** Создайте рабочую директорию для установки.

1. Создайте директорию для развертывания:

```
sudo mkdir -p <DEPLOYMENT_PATH_SERVER>
```

Пример:

```
sudo mkdir -p /opt/alterofficeweb/
```

2. Перейдите в директорию развертывания:

```
cd <DEPLOYMENT_PATH_SERVER>
```

Пример:

```
cd /opt/alterofficeweb/
```

**Шаг 5.** Если доменное имя не определяется через DNS, укажите его соответствие IP-адресу в файле /etc/hosts.

```
sudo bash -c "echo ' <SERVER_IP> <SERVER_HOSTNAME>' >> /etc/hosts"
```

Пример:

```
sudo bash -c "echo '172.20.18.10 demo03-web2025.alteroffice.ru' >> /etc/hosts"
```

**Шаг 6.** Возьмите следующие файлы из репозитория <https://repo.alteroffice.ru/web/> любым удобным способом:

- Архив с установщиком alterofficeweb-installer\_\*.zip
- Набор Docker-образов alterofficeweb-bundle\_\*.tar

```
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: 8040323072 (7,5G) [application/x-tar]
Сохранение в: «alterofficeweb-bundle_v2025.0.0-8.tar»
_v2025.0.0-8.tar          69%[=====]          ] 5,20G 8,33MB/s    Ост 4м 14s
```

Рисунок 3. Скачивание файлов из репозитория

**Шаг 7.** Распакуйте архив с установщиком в текущую директорию:

```
sudo unzip alterofficeweb-installer_*.zip
```

```
[.../x alteroffice]$ ls
alterofficeweb-bundle_v2025.0.0-8.tar  alterofficeweb-installer_v2025.0.0-8.zip
[.../x alteroffice]$ sudo unzip alterofficeweb-installer_v2025.0.0-8.zip
Archive: alterofficeweb-installer_v2025.0.0-8.zip
  inflating: install.sh
  inflating: compose.yaml
  inflating: compose-elasticsearch.yaml
[.../x alteroffice]$ ls
alterofficeweb-bundle_v2025.0.0-8.tar      compose-elasticsearch.yaml  install.sh
alterofficeweb-installer_v2025.0.0-8.zip  compose.yaml
```

Рисунок 4. Извлечение файлов из архива

**Шаг 8.** Подготовьте SSL сертификаты, которые будут использоваться для HTTPS доступа к серверу:

2. Создайте директорию для SSL сертификатов :

```
sudo mkdir -p <PATH_SSL>
```

Пример:

```
sudo mkdir -p /etc/ssl/alteroffice
```

2. Поместите в директорию SSL сертификаты для вашего домена:

- **server.crt** - SSL сертификат
- **server.key** - приватный ключ

3. Убедитесь, что файлы имеют правильные права доступа:

```
sudo chmod 644 <PATH_SSL>/server.crt
sudo chmod 600 <PATH_SSL>/server.key
```

Пример:

```
sudo chmod 644 /etc/ssl/alteroffice/server.crt
sudo chmod 600 /etc/ssl/alteroffice/server.key
```

#### Внимание

Убедитесь, что доменное имя указано правильно и DNS запись ведет на ваш сервер.

Директория для контента будет создана автоматически скриптом.

Директория для SSL сертификатов должна существовать до запуска инсталляционного скрипта.

```
~]$ ls -la /etc/ssl/alteroffice/
итого 8
drwxr-xr-x  2 root root  42 окт 31 14:58 .
drwxr-xr-x. 3 root root  96 окт 31 14:58 ..
-rw-r--r--  1 root root 1830 окт 31 14:58 server.crt
-rw-----  1 root root 3272 окт 31 14:58 server.key
[~]$ sudo chmod 644 /etc/ssl/alteroffice/server.crt
sudo chmod 600 /etc/ssl/alteroffice/server.key
[~]$ ls -la /etc/ssl/alteroffice/
итого 8
drwxr-xr-x  2 root root  42 окт 31 14:58 .
drwxr-xr-x. 3 root root  96 окт 31 14:58 ..
-rw-r--r--  1 root root 1830 окт 31 14:58 server.crt
-rw-----  1 root root 3272 окт 31 14:58 server.key
[~]$ cd alteroffice/
[alteroffice]$ ls -la
итого 7851916
drwxr-xr-x  2 root root      171 окт 31 14:57 .
drwx----- 18 root root    4096 окт 31 14:41 ..
-rw-r--r--  1 root root 8040323072 окт 31 13:53 alterofficeweb-bundle_v2025.0.0-8.tar
-rw-r--r--  1 root root      5050 окт 31 13:53 alterofficeweb-installer_v2025.0.0-8.zip
-rw-rw-r--  1 root root      907 окт 31 13:53 compose-elasticsearch.yaml
-rw-rw-r--  1 root root    2810 окт 31 13:53 compose.yaml
-rwxrwxr-x  1 root root   12547 окт 31 13:44 install.sh
```

Рисунок 5. Подготовка сертификатов

**Шаг 9.** Убедитесь, что в директории установки появились необходимые файлы:

- **compose.yaml**
- **compose-elasticsearch.yaml**

- `install.sh`

---

## 4.2. Установка

### Шаг 1. Настройка внешнего S3-хранилища при развертывании с нуля

Если вы хотите подключить внешнее S3 хранилище Minio до развертывания системы, то обратитесь к инструкции, см. раздел **9.2**.

### Шаг 2. Запустите скрипт установки:

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

### Шаг 3. Следуйте инструкциям скрипта:

- Укажите директорию для хранения контента (например: `/var/www/alteroffice`).
- Укажите директорию с SSL сертификатами (например: `/etc/ssl/alteroffice`).
- Введите доменное имя (например: `office.example.com`).
- Подтвердите автоматически определенный IP или введите вручную.
- Введите логин администратора системы АльтерОфис Веб (по умолчанию: `admin`).
- Введите пароль администратора АльтерОфис Веб.

```
[root@alteroffice alteroffice]$ sudo ./install.sh
== Установка AlterOfficeWeb ==
Docker установлен: Docker version 20.10.25, build b82b9f3
Найден Docker Compose плагин: Docker Compose version 2.26.1
Введите директорию для хранения контента: /var/www/alteroffice
Введите директорию где находятся SSL сертификаты: /etc/ssl/alteroffice
Введите доменное имя: office.example.com
Введите логин администратора AlterOffice [admin]: admin
Введите пароль администратора AlterOffice:
Сгенерирован случайный пароль для PostgreSQL
Создана директория: /var/www/alteroffice
RSA key ok
SSL сертификаты найдены и валидны: /etc/ssl/alteroffice/server.crt и /etc/ssl/alteroffice/server.key
Найден бандл: ./alterofficeweb-bundle_v2025.0.0-8.tar
Загрузка docker образов из ./alterofficeweb-bundle_v2025.0.0-8.tar...
```

Рисунок 6. Процесс установки

### Шаг 4. Дождитесь выполнения скрипта.

По завершению работы скрипта будет выведено уведомление "== AlterOfficeWeb запущен ==".

### Шаг 5. Проверьте список контейнеров:

```
sudo docker ps
```

```
[sudo] пароль для root:
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
e83e2ff58965      alterofficeweb/front:v2025.0.0-8   "/entrypoint.sh"   11 minutes ago   Up 10 minutes   80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp   prod-front-1
131b54ed799b      alterofficeweb/app:v2025.0.0-8     "/cron.sh"        11 minutes ago   Up 11 minutes   9000/tcp           prod-cron-1
0c46198cdd91      alterofficeweb/docs:v2025.0.0-8   "/entrypoint.sh"   11 minutes ago   Up 11 minutes   80/tcp              prod-app-1
16a742fb705      alterofficeweb/postgres:16.9-alpine3.22  "/docker-entrypoint.s_  11 minutes ago   Up 11 minutes   5432/tcp           prod-docs-1
228c335a7883      alterofficeweb/redis:7.4.5-alpine3.21  "/docker-entrypoint.s_  11 minutes ago   Up 11 minutes   6379/tcp           prod-db-1
7b370c4bc799      alterofficeweb/editors-online:v2025.0.0-8  "/start-collabora-on_  11 minutes ago   Up 11 minutes   9900/tcp           prod-editors-1
bf7d2042040b      alterofficeweb/kibana:8.11.4        "/bin/tini -- /usr/l_  11 minutes ago   Up 11 minutes   0.0.0.0:5601->5601/tcp, :::5601->5601/tcp   prod-elk-kibana-1
ff1d18a8e264b      alterofficeweb/elasticsearch:v2025.0.0-8  "/bin/tini -- /usr/l_  11 minutes ago   Up 11 minutes   9200/tcp, 9300/tcp           prod-elk-elasticsearch-1
4ae0924b25c5      alterofficeweb/elasticsearch:v2025.0.0-8  "/bin/tini -- /usr/l_  11 minutes ago   Up 11 minutes   9200/tcp, 9300/tcp           prod-elk-elasticsearch-1

[centos@centos alteroffice]$
```

Рисунок 7. Список контейнеров после установки

#### Шаг 6. Проверьте доступность веб-интерфейса.

Через несколько минут после завершения установки интерфейс «Альтер Офис Веб» становится доступным. Получить к нему доступ можно через веб браузер, используя имя сервера и данные учётной записи, заданные при установке.

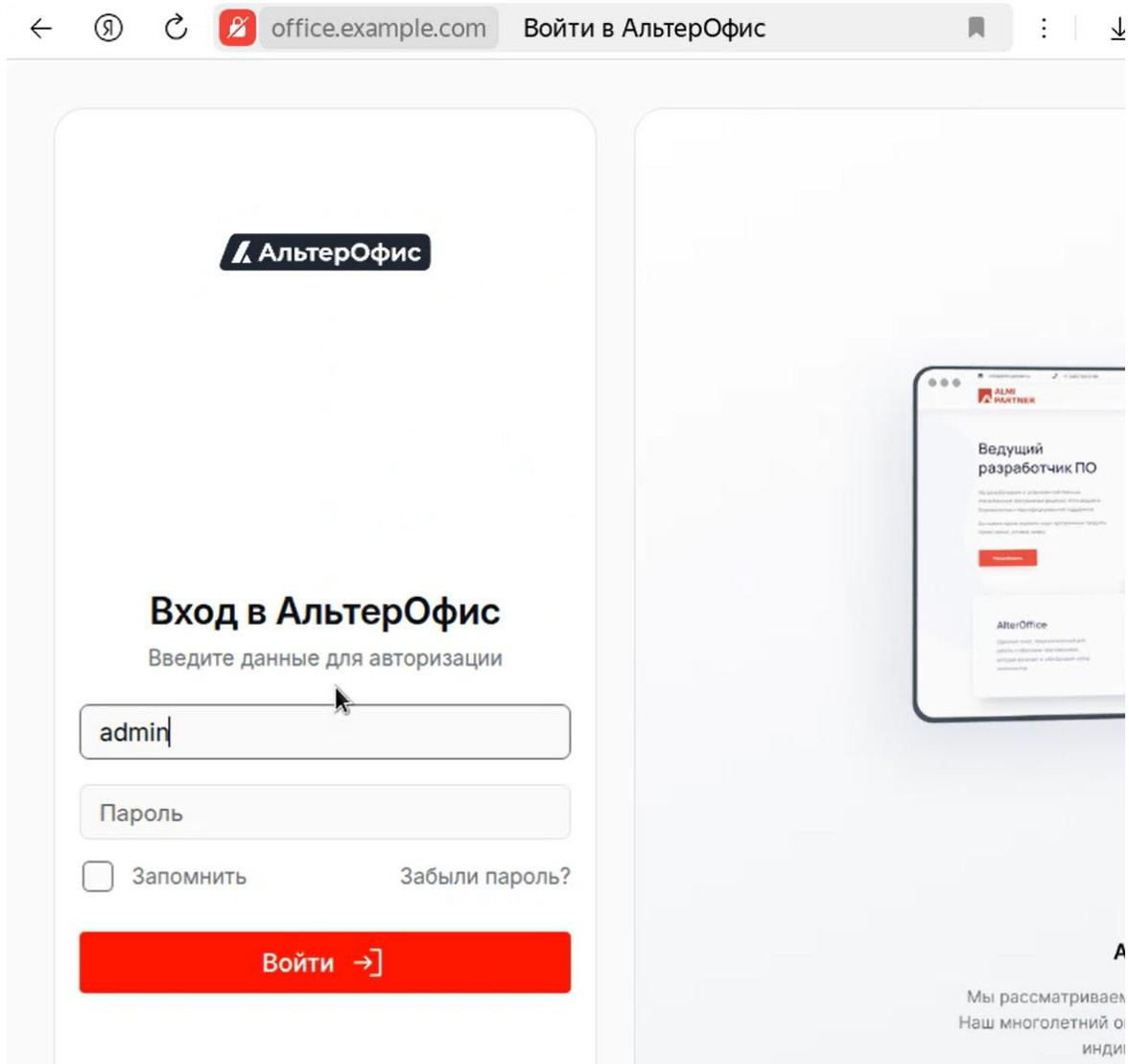


Рисунок 8. Окно авторизации АльтерОфис Веб

#### Шаг 7. Подключение внешнего S3 хранилища на уже развернутой системе.

Если вы хотите подключить внешнее S3 хранилище Minio на уже развернутой системе, то обратитесь к инструкции, см. раздел **9.1**.

---

## 4.3. Настройка системы для работы с макросами

По умолчанию макросы в редакторах отключены в целях безопасности.

Работа с макросами не является обязательным при установке системы и настраивается при необходимости. Настройки выполняются через командную строку (CLI) и через графический веб-интерфейс (WebUI).

---

### 4.3.1. Порядок выполнения настроек для работы с макросами

Для успешной настройки рекомендуется соблюдать следующий порядок:

Настройка	Вариант настройки
Администратор системы настраивает контейнеры <b>app</b> и <b>editors</b> для работы с одним <b>VOLUME</b> , в который будут размещаться макросы.	CLI
Администратор системы подключает внешнее хранилище для размещения в нем макросов приложения через WebUI и которые будут доступны пользователям.	WebUI
Администратор системы создает новые макросы или загружает ранее созданные макросы в хранилище макросов.	WebUI
Пользователи АльтерОфис Веб используют настроенные администратором системы макросы для своей работы в офисных редакторах (АТекст, АТаблица, АКонцепт).	WebUI

Администратор системы может настроить ограниченные права доступа к системе, достаточные для создания и размещения макросов приложения, и предоставить их пользователю, который будет отвечать за макросы.

Выполнение настроек для работы с макросами в системе «АльтерОфис Веб» будет рассмотрено на примере:

- Экземпляр имеет адрес <https://demo03-web2025.alteroffice.ru>

Параметры, которые потребуются для настройки:

Параметр	Описание	Пример
<b>SERVER_URL</b>	URL для доступа к системе АльтерОфис Веб	<code>https://demo03-web2025.alteroffice.ru</code>
<b>PATH_EDITORS</b>	Путь к папке в контейнере <b>editors</b> , где будут размещаться макросы приложения	<code>/opt/lokit/share/Scripts/python</code>
<b>PATH_APP</b>	Путь к папке в контейнере <b>app</b> , где будут размещаться макросы приложения	<code>/var/www/html/data/Scripts</code>
<b>PATH_VOLUME</b>	Путь к <b>VOLUME</b> на хосте, где будут размещаться макросы приложения	<code>/opt/alterofficeweb/demo03/html/data/Scripts</code>
<b>PATH_SERVER</b>	Путь к каталогу, где установлена система.	<code>/opt/alterofficeweb/demo03</code>

#### 4.3.2. Подготовка yaml-файлов

Для корректной совместной работы контейнеров **app** и **editors** с макросами, необходимо настроить общий том, смонтировав локальную директорию хоста в соответствующие пути внутри каждого контейнера.

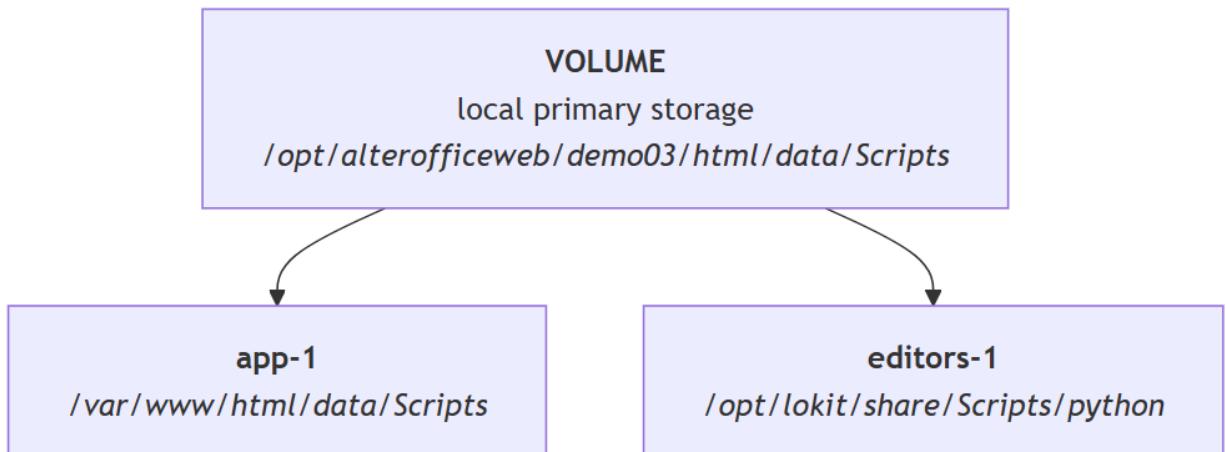


Рисунок 9. Схема общего доступа к макросам через хост

**Шаг 1.** Перейдите в каталог, где установлена система.

`cd <PATH_SERVER>`

Пример:

`cd /opt/alterofficeweb/demo03`

**Шаг 2.** Подготовьте обновленный **compose.yaml** файл для контейнера **app** настраиваемого стенда (например, **demo03**).

Откройте файл на редактирование:

```
nano compose.yaml
```

Внесите изменения.

```
redis:
  ...
app:
  image: ...
  restart: always
  volumes:
    - ...
    - <PATH_VOLUME>:<PATH_APP>
  ...
  ...
```

Замените:

- <PATH\_VOLUME> на путь к директории на хосте, где будут размещаться макросы;
- <PATH\_APP> на путь в контейнере **app**, где будут размещаться макросы.

Пример:

```
redis:
  ...
app:
  image: ...
  restart: always
  volumes:
    - ...
    - /opt/alterofficeweb/dev02/html/data/Scripts:/var/www/html/data/Scripts
  ...
  ...
```

Сохраните изменения и выйдите из редактора.

**Шаг 3.** Подготовьте обновленный **compose-editors.yaml** файл для контейнера **editors** настраиваемого стенда (например, **demo03**).

Откройте файл на редактирование:

```
nano compose-editors.yaml
```

Внесите изменения.

```
editors:
  image: ...
  restart: always
  volumes:
```

```
- <PATH_VOLUME>:<PATH_EDITORS>
...
```

Замените:

- <PATH\_VOLUME> на путь к директории на хосте, где будут размещаться макросы;
- <PATH\_EDITORS> на путь в контейнере **editors**, где будут размещаться макросы.

Пример:

```
editors:
  image: ...
  restart: always
  volumes:
    - /opt/alterofficeweb/demo03/html/data/Scripts:/opt/lokit/share/Scripts/python
...

```

Сохраните изменения и выйдите из редактора.

---

#### 4.3.3. Смена владельца для VOLUME

Чтобы администратор системы мог размещать макросы в системе, необходимо корректно настроить владельца для созданной папки на хосте (**VOLUME**):

```
chown -R <пользователь>:<группа> <PATH_VOLUME>
```

Пример:

```
chown -R 33:33 /opt/alterofficeweb/demo03/html/data/Scripts
```

---

#### 4.3.4. Включение макросов

По умолчанию в редакторах макросы отключены. Чтобы их включить, нужно настроить файл coolwsd.xml, включив макросы и изменив уровень безопасности:

```
<enable_macros_execution desc="Specifies whether the macro execution is enabled in general. This will enable Basic, Beanshell, Javascript and Python scripts. If it is set to false, the macro_security_level is ignored. If it is set to true, the mentioned entry specified the level of macro security." type="bool" default="false">true</enable_macros_execution>
```

Чтобы включить выполнение макросов, необходимо изменить значение параметра **enable\_macros\_execution** с **false** на **true**.

Также, необходимо выбрать уровень безопасности, которому следует придерживаться при выполнения макроса.

Существует два уровня:

- 0 (Низкий, не рекомендуется) - Все макросы будут выполняться без подтверждения.
- 1 (Средний, по умолчанию) - Требуется подтверждение перед выполнением макросов из ненадежных источников.

```
<macro_security_level desc="Level of Macro security. 1 (Medium) Confirmation required before executing macros from untrusted sources. 0 (Low, not recommended) All macros will be executed without confirmation." type="int" default="1">1</macro_security_level>
```

**Шаг 1.** Перейдите в каталог, где установлена система.

```
cd <PATH_SERVER>
```

Пример:

```
cd /opt/alterofficeweb/demo03
```

**Шаг 2.** Откройте файл **.env-editors** на редактирование.

В файле **.env-editors** настраиваются переменные окружения, которые прокидываются в файл coolwsd.xml контейнера **editors**.

Пример:

```
nano /opt/alterofficeweb/demo03/.env-editors
```

**Шаг 3.** Добавьте параметры **--o:security.enable\_macros\_execution=true** **--o:security.macro\_security\_level=0** в конце существующих настроек.

Пример:

```
domain=demo03-web2025.alteroffice.ru
extra_params=--o:ssl.enable=true --o:ssl.termination=false --o:net.lok_allow.host[14]=demo03-web2025.alteroffice.ru --o:security.enable_macros_execution=true --o:security.macro_security_level=0
```

---

#### 4.3.5. Применение внесенных изменений

**Шаг 1.** Перезапустите контейнер **app**

```
docker compose restart app
```

**Шаг 2.** Примените настройки к контейнеру **editors** и проверьте применение параметров

```
docker compose -f compose-editors.yaml down
```

Команда очистит окружение (остановит и удалит запущенные контейнеры, описанные в **compose-editors.yaml**).

```
docker compose -f compose-editors.yaml up -d
```

Команда запустит все сервисы, описанные в файле **compose-editors.yaml**, в фоновом режиме (демоне).

**Шаг 3.** Проверьте, что переменные корректно передались в контейнер:

```
docker exec demo03-editors-1 printenv
```

```
[root@localhost de... ]# docker exec de... l-editors-1 printenv
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=5d1ff02b4d6c
extra_params=--o:ssl.enable=true --o:ssl.termination=false --o:security.enable_macros_execution=true --o:security.macro_security_level=0
domain=de...-web2025.alteroffice.ru
HOME=/opt/cool
[root@localhost de... ]#
```

Рисунок 10. Проверка настройки переменных окружения

**Шаг 4.** Проверьте доступность кнопки «Выполнить макрос» в онлайн-редакторах.

После применения обновленных настроек файла **coolwsd.xml** проверьте, что в интерфейсе появилась кнопка «Выполнить макрос».

Для проверки, зайдите в АльтерОфис Веб, откройте АТаблица.

Убедитесь, что на ленте на вкладке «Файл» появилась кнопка «Выполнить макрос».

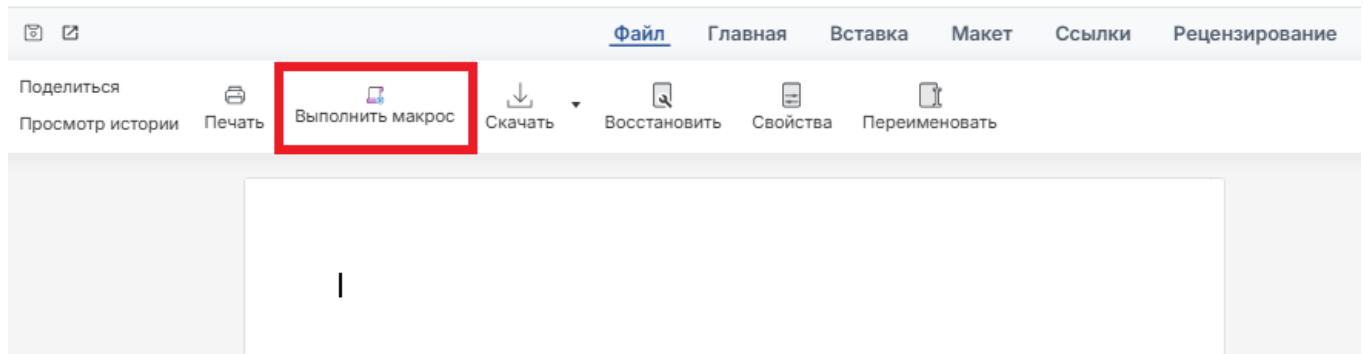


Рисунок 11. Кнопка «Выполнить макрос» активна.

#### 4.3.6. Загрузка и создание макросов в АльтерОфис Веб

Дальнейшая настройка АльтерОфис Веб для работы с макросами выполняется через веб-интерфейс.

Описание шагов по дальнейшей настройке системы с использованием веб-интерфейса описаны в документе **«Руководство по настройке и администрированию АльтерОфис Веб»**:

- Подключение внешнего хранилища для работы с макросами.
- Загрузка и создание макросов.
- Работа пользователей с макросами приложения.

## 4.4. Настройка федеративного обмена между серверами

Между несколькими экземплярами АльтерОфис Веб может быть настроен федеративный доступ, реализованный на протоколе Open Cloud Mesh (OCM).

Каждый экземпляр функционирует автономно, но при необходимости пользователи могут обмениваться файлами и работать над документами совместно, как если бы они находились на одном сервере.

Федеративный доступ не является обязательным при установке системы и настраивается при необходимости. Настройки выполняются через командную строку (CLI) и через графический веб-интерфейс (WebUI).

В данном разделе описывается часть по настройкам с использованием CLI, настройки с использованием WebUI описаны в документе **«Руководство по настройке и администрированию АльтерОфис Веб»**.

Настройка федеративного обмена файлами между двумя серверами АльтерОфис Веб, развернутыми в Docker-контейнерах, будет рассмотрена на примере двух экземпляров:

- **Экземпляр 1** имеет адрес <https://demo03-web2025.alteroffice.ru>
- **Экземпляр 2** имеет адрес <https://demo04-web2025.alteroffice.ru>

Параметры, которые потребуются для настройки:

Параметр	Описание	Пример
<b>SOURCE_SERVER_URL</b>	URL исходного сервера (источник файлов), <b>Экземпляр 1</b>	<a href="https://demo03-web2025.alteroffice.ru">https://demo03-web2025.alteroffice.ru</a>
<b>DESTINATION_SERVER_URL</b>	URL целевого сервера (получатель файлов), <b>Экземпляр 2</b>	<a href="https://demo04-web2025.alteroffice.ru">https://demo04-web2025.alteroffice.ru</a>
<b>SOURCE_SERVER_NAME</b>	Название директории установки АльтерОфис Веб на исходном сервере, <b>Экземпляр 1</b>	demo03
<b>DESTINATION_SERVER_NAME</b>	Название директории установки АльтерОфис Веб на целевом сервере, <b>Экземпляр 2</b>	demo04
<b>SOURCE_SERVER_HOSTNAME</b>	Доменное имя или FQDN (Fully Qualified Domain Name) исходного сервера АльтерОфис Веб, с которого будут предоставляться общие	<a href="https://demo03-web2025.alteroffice.ru">demo03-web2025.alteroffice.ru</a>

Параметр	Описание	Пример
<b>DESTINATION_SERVER_HOSTNAME</b>	папки и файлы для федеративного обмена.	
<b>SOURCE_SERVER_IP</b>	Доменное имя или FQDN целевого сервера, на котором пользователи будут получать доступ к общим ресурсам с исходного сервера.	demo04-web2025.alteroffice.ru
<b>DESTINATION_SERVER_IP</b>	IP-адрес хоста исходного сервера, где установлен контейнер <b>app</b> для <b>Экземпляр 1</b>	172.20.18.10
<b>DEPLOYMENT_PATH_SOURCE_SERVER</b>	— IP-адрес хоста целевого сервера, где установлен контейнер <b>app</b> для <b>Экземпляр 2</b>	172.45.58.20
<b>DEPLOYMENT_PATH_DESTINATION_SERVER</b>	Путь к директории развертывания на хосте (путь к файлу compose.yaml для <b>Экземпляр 1</b> )	/opt/alterofficeweb/demo 03
<b>SERVICE_USER_SOURCE_SERVER</b>	Путь к директории развертывания на хосте (путь к файлу compose.yaml для <b>Экземпляр 2</b> )	/opt/alterofficeweb/demo 04
<b>SERVICE_USER_DESTINATION_SERVER</b>	Пользователь для развертывания и управления контейнерами <b>Экземпляр 1</b>	gitlab-runner
<b>SERVICE_USER_DESTINATION_SERVER</b>	Пользователь для развертывания и управления контейнерами <b>Экземпляр 2</b>	gitlab-runner

#### 4.4.1. Порядок выполнения операций для настройки федеративного доступа

Для успешной настройки рекомендуется соблюдать следующий порядок:

Настройка	Вариант настройки
Настройка сетевого взаимодействия	CLI
Проверка доступности	CLI
Активация приложения Federation	CLI

Настройка	Вариант настройки
Разрешение доступа для изолированных серверов	CLI
Установка базовых URL	CLI
Настройка «белого» списка IP адресов	WebUI
Настройка межсерверного обмена для пользователей	WebUI
Настройка доверенных серверов	WebUI
Синхронизация адресных книг для федеративного доступа	CLI
Настройка редакторов для совместной работы при федеративном доступе	CLI

#### 4.4.2. Настройка сетевого взаимодействия

Для обеспечения корректного разрешения имен между серверами необходимо добавить статические записи в файл хостов внутри каждого контейнера.

**На исходном сервере (SOURCE\_SERVER\_URL):**

**Шаг 1.** Перейдите в директорию развертывания:

```
cd <DEPLOYMENT_PATH_SOURCE_SERVER>/<SOURCE_SERVER_NAME>
```

Пример:

```
cd /opt/alterofficeweb/dev03
```

**Шаг 2.** Отредактируйте файл compose.yaml:

```
sudo -u <SERVICE_USER_SOURCE_SERVER> nano compose.yaml
```

Пример:

```
sudo -u gitlab-runner nano compose.yaml
```

**Шаг 3.** В секции сервиса app добавьте или измените блок extra\_hosts:

```
services:
  app:
    # ... существующие настройки
    extra_hosts:
      - "<SOURCE_SERVER_HOSTNAME>:<SOURCE_SERVER_IP>"
      - "<DESTINATION_SERVER_HOSTNAME>:<DESTINATION_SERVER_IP>"
```

Пример:

```
extra_hosts:
  - "demo03-web2025.alteroffice.ru:172.20.18.10"
  - "demo04-web2025.alteroffice.ru:172.45.58.20"
```

Примечание

При редактировании файла **compose.yaml** не забывайте про требования к оформлению YAML-файлов:

Для отступов необходимо использовать пробелы, а не табуляции.

Обычно на каждый уровень отступа приходится 2 пробела.

**Шаг 4.** Сохраните изменения и перезапустите контейнер:

```
docker compose restart app
```

**На целевом сервере (DESTINATION\_SERVER\_URL):**

Повторите те же шаги, указав соответствующие параметры:

`extra_hosts:`

- "<DESTINATION\_SERVER\_HOSTNAME>:<DESTINATION\_SERVER\_IP>"
- "<SOURCE\_SERVER\_HOSTNAME>:<SOURCE\_SERVER\_IP>"

Пример:

`extra_hosts:`

- "demo04-web2025.alteroffice.ru:172.45.58.20"
- "demo03-web2025.alteroffice.ru:172.20.18.10"

#### 4.4.3. Проверка сетевой доступности

После настройки DNS-записей необходимо убедиться, что серверы могут взаимодействовать друг с другом без SSL-ошибок.

##### Проверка с исходного сервера

Выполните команду для проверки доступности целевого сервера:

```
docker exec -it <SOURCE_CONTAINER_NAME> curl <DESTINATION_SERVER_URL>/status.php
```

Пример:

```
docker exec -it demo03-app-1 curl https://demo04-web2025.alteroffice.ru/status.php
```

##### Примечание

Команда curl выполняется внутри контейнера app без входа в сам контейнер.

Ожидаемый результат:

```
{"installed":true,"maintenance":false,"needsDbUpgrade":false,"version":"31.0.4.17227","versionstring":"2025.0.0.9","edition":"","productname":"AlterOffice","extendedSupport":false}
```

##### Проверка с целевого сервера

Аналогично проверьте доступность исходного сервера:

```
docker exec -it <DESTINATION_CONTAINER_NAME> curl <SOURCE_SERVER_URL>/status.php
```

Пример:

```
docker exec -it demo04-app-1 curl https://demo03-web2025.office.ru/status.php
```

Если возникают SSL-ошибки, убедитесь, что сертификаты на обоих серверах корректно настроены и доверены.

---

#### 4.4.4. Активация приложения Federation

Для работы федеративного обмена необходимо активировать соответствующее приложение на обоих серверах.

На каждом сервере выполните:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ app:enable federation
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ app:enable federation
```

Пример для целевого сервера:

```
docker exec -it demo04-app-1 php /var/www/html/occ app:enable federation
```

Возможные результаты выполнения:

При успешной активации:

- federation 1.21.0 enabled

Если приложение уже активировано:

- federation already enabled

Использование WebUI для управления приложениями см. в документе «**Руководство по настройке и администрированию АльтерОфис Веб**».

---

#### 4.4.5. Настройка доступа для изолированных серверов

По умолчанию АльтерОфис Веб разрешает федеративный обмен только с серверами, имеющими валидные публичные SSL-сертификаты. Для работы в изолированном контуре необходимо разрешить обмен с локальными серверами.

На каждом сервере выполните:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ config:system:set allow_local_remote_servers --value=true
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:system:set allow_local_remote_servers --value=true
```

Пример для целевого сервера:

```
docker exec -it demo04-app-1 php /var/www/html/occ config:system:set allow_local_remote_servers --value=true
```

Результат выполнения:

```
System config value allow_local_remote_servers set to boolean true
```

---

#### 4.4.6. Установка базового URL-адреса

Для корректной генерации абсолютных ссылок в системе (например, в уведомлениях по электронной почте или для внешних пользователей) необходимо явно задать базовый URL каждого сервера.

**На исходном сервере выполните:**

```
docker exec -it <SOURCE_CONTAINER_NAME> php /var/www/html/occ config:system:set overwrite.cli.url --value='<SOURCE_SERVER_URL>'
```

Пример:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:system:set overwrite.cli.url --value='https://demo03-web2025.office.ru'
```

**На целевом сервере выполните:**

```
docker exec -it <DESTINATION_CONTAINER_NAME> php /var/www/html/occ config:system:set overwrite.cli.url --value='<DESTINATION_SERVER_URL>'
```

Пример:

```
docker exec -it demo04-app-1 php /var/www/html/occ config:system:set overwrite.cli.url --value='https://demo04-web2025.office.ru'
```

Результат выполнения:

```
System config value overwrite.cli.url set to string <URL>
```

Параметр будет добавлен в файл конфигурации config/config.php:

```
'overwrite.cli.url' => '<URL> ',
```

---

#### 4.4.7. Настройка «белого» списка IP адресов

Для корректной работы серверов при организации федеративного доступа, необходимо IP адреса серверов добавить в «белый список».

Настройка «белого списка» выполняется через веб-интерфейс и описано в документе **«Руководство по настройке и администрированию АльтерОфис Веб»**.

---

#### 4.4.8. Настройка межсерверного обмена для пользователей

Активируйте необходимые опции:

Параметр	Назначение	Примечание
<b>Разрешить пользователям на этом сервере публиковать общие ресурсы на других серверах (этот параметр также разрешает доступ WebDAV к общим папкам)</b>	Разрешает пользователям данного сервера предоставлять доступ к своим файлам и папкам пользователям на других (внешних) серверах.	Используйте, если требуется функционал совместной работы с внешними организациями или другими экземплярами АльтерОфис Веб.
<b>Разрешить пользователям этого сервера принимать общие ресурсы с других серверов</b>	Разрешает пользователям данного сервера получать и принимать файлы и папки, которыми с ними поделились пользователи внешних серверов.	Включите для полноценного двустороннего обмена. Обычно активируется вместе с предыдущей опцией.
<b>Разрешить пользователям этого сервера предоставлять общий доступ группам пользователей других серверов</b>	Позволяет пользователям этого сервера предоставлять доступ целым группам пользователей, существующим на внешнем сервере. Удобно для совместной работы с отделами или командами в другой организации.	Экспериментальная функция.
<b>Разрешить пользователям этого сервера принимать общие ресурсы с других серверов, опубликованные для групп пользователей</b>	Разрешает группам пользователей на этом сервере получать общие ресурсы от пользователей внешних серверов. Все члены группы получат доступ к присланным файлам/папкам.	Экспериментальная функция.

Параметр	Назначение	Примечание
<b>По умолчанию автоматически принимать общие ресурсы от доверенных федеративных учетных записей и групп</b>	Автоматически принимает входящие общие ресурсы от серверов и групп, добавленных в «доверенные», без необходимости ручного подтверждения каждым пользователем.	Включайте только для серверов-партнёров с высоким уровнем доверия (например, внутри холдинга).

## На исходном сервере

Выполните настройку на исходном сервере.

- 1. Включение опции «Разрешить пользователям на этом сервере публиковать общие ресурсы на других серверах (этот параметр также разрешает доступ WebDAV к общим папкам)»**

Опция позволяет разрешать пользователям отправлять файлы на другие серверы.

Проверьте текущий статус:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:get files_sharing outgoing_server2server_share_enabled
```

Для выключения выполните:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing outgoing_server2server_share_enabled --value="no"
```

Для включения:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing outgoing_server2server_share_enabled --value="yes"
```

- 2. Включение опции «Разрешить пользователям этого сервера принимать общие ресурсы с других серверов»**

Опция позволяет пользователям принимать файлы с других серверов.

Проверить текущий статус:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:get files_sharing incoming_server2server_share_enabled
```

Выключить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing incoming_server2server_share_enabled --value="no"
```

Включить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing incoming_server2server_share_enabled --value="yes"
```

- 3. Включение опции «Разрешить пользователям этого сервера предоставлять общий доступ группам пользователей других серверов»

Экспериментальная функция.

Проверить текущий статус:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:get files_sharing outgoing_server2server_group_share_enabled
```

Выключить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing outgoing_server2server_group_share_enabled --value="no"
```

Включить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing outgoing_server2server_group_share_enabled --value="yes"
```

- 4. Включение опции «Разрешить пользователям этого сервера принимать общие ресурсы с других серверов, опубликованные для групп пользователей»

Экспериментальная функция.

Проверить текущий статус:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:get files_sharing incoming_server2server_group_share_enabled
```

Выключить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing incoming_server2server_group_share_enabled --value="no"
```

Включить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing incoming_server2server_group_share_enabled --value="yes"
```

- 5. Включение опции «По умолчанию автоматически принимать общие ресурсы от доверенных федеративных учетных записей и групп»

Проверить текущий статус:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:get files_sharing federatedTrustedShareAutoAccept
```

Выключить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing  
federatedTrustedShareAutoAccept --value="no"
```

Включить:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set files_sharing  
federatedTrustedShareAutoAccept --value="yes"
```

#### На целевом сервере

Аналогично выполните настройки на целевом сервере.

См. также

Настройка межсерверного обмена с использованием веб-интерфейса описано в документе **«Руководство по настройке и администрированию АльтерОфис Веб»**.

#### 4.4.9. Настройка доверенных серверов

Настройка доверенных серверов выполняется с использованием веб-интерфейса, см. описание в документе **«Руководство по настройке и администрированию АльтерОфис Веб»**.

Для ускорения процесса добавления сервера в доверенные должны отработать фоновые задания:

- OCA\Federation\BackgroundJob\GetSharedSecret
- OCA\Federation\BackgroundJob\RequestSharedSecret

Фоновое задание **GetSharedSecret** используется для получения и верификации общего секретного ключа от удалённого сервера при установлении доверенного соединения.

Фоновое задание **RequestSharedSecret** используется для инициирования запроса на создание общего секрета к удалённому серверу. Запускает процесс рукопожатия (handshake) перед началом федеративного обмена, отправляя запрос на установление защищённого канала связи.

Для проверки статуса воспользуйтесь командой:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ background-job:list | grep  
p Secret
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ background-job:list | grep Se  
cret
```

В результате будет выведен список заданий с идентификаторами.

Для принудительного запуска фонового задания используйте полученный идентификатор в команде выполнения:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ background-job:execute <JOB_ID>
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ background-job:execute 312>
```

После выполнения фоновых заданий GetSharedSecret и RequestSharedSecret, должно быть выполнено задание SyncJob.

Фоновое задание **SyncJob** используется для периодической синхронизации списка пользователей и групп с доверенными федеративными серверами.

Запуск задания **SyncJob** является обязательным для завершения процесса добавления сервера в доверенные. Только после этого статус индикатор изменится на зеленый и будет осуществлена синхронизация адресных книг, которая позволяет искать пользователей с другого сервера.

Определите идентификатор задания выполнив команду:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ background-job:list | grep SyncJob
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ background-job:list | grep SyncJob
```

Результат выполнения:

```
| 12 | OCA\Federation\SyncJob | 2025-12-05T07:42:21+00:00 | null |
```

Для принудительного запуска фонового задания выполните команду:

```
# Запускаем принудительно задание
docker exec -it <CONTAINER_NAME> php /var/www/html/occ background-job:execute <JOB_ID> --force-execute
```

Пример для исходного сервера:

```
# Запускаем принудительно задание
docker exec -it demo03-app-1 php /var/www/html/occ background-job:execute 12 --force-execute
```

После выполнения команды индикатор доверенного сервера в веб-интерфейсе должен стать зелёным.

## Доверенные серверы ①

Федерация позволяет вам подключаться к другим доверенным серверам для обмена каталогом учетных записей. Например, это будет использоваться для автоматического заполнения внешних учетных записей для федеративного общего доступа. Для создания федеративного общего ресурса нет необходимости добавлять сервер в качестве доверенного.

Каждый сервер должен проверить другой. Этот процесс может потребовать нескольких циклов cron.

● <https://dev03-web2025.alteroffice.ru>

[+ Добавить доверенный сервер](#)

Повторите аналогичные действия для целевого сервера.

---

### 4.4.10. Синхронизация адресных книг для федеративного доступа

Синхронизация адресных книг обеспечивает автоматический обмен контактами между доверенными серверами, позволяя пользователям быстро находить коллег на партнёрских экземплярах АльтерОфис Веб при предоставлении общего доступа. Эта функция работает в фоновом режиме через cron задания и активируется при успешном установлении «зелёного» статуса соединения между серверами.

Для ускорения процесса синхронизации выполните команду:

```
docker exec -it <CONTAINER_NAME> php /var/www/html/occ federation:sync-addressbooks
```

Пример для исходного сервера:

```
docker exec -it demo03-app-1 php /var/www/html/occ federation:sync-addressbooks
```

Для целевого сервера выполните аналогичные действия.

---

### 4.4.11. Настройка редакторов для совместной работы при федеративном доступе

Для открытия офисных документов, к которым предоставлен федеративный доступ, необходимо в контейнере редактора **editors** выполнить настройки.

**Для исходного сервера**

- **Для контейнера editors:**

```
docker exec -it demo03-editors-1 coolconfig set net.content_security_policy "frame-ancestors *.alteroffice.ru: *;" --config=/etc/coolwsd/coolwsd.xml
```

Ожидаемый результат выполнения:

```
Previous value found in config file: ""
Changing value to: "frame-ancestors *.alteroffice.ru:/*"
Saving configuration to : /etc/coolwsd/coolwsd.xml ...
Saved
```

- Для контейнера app:

```
docker exec -it demo03-app-1 php /var/www/html/occ config:app:set richdocuments
federation_use_trusted_domains --value="yes"
```

Ожидаемый результат выполнения:

```
Config value 'federation_use_trusted_domains' for app 'richdocuments' is now set
to 'yes', stored as mixed in fast cache
```

Для целевого сервера выполните аналогичные действия.

---

## 5. Резервное копирование и восстановление

Резервное копирование и восстановление системы «АльтерОфис Веб» будет рассмотрено на примере:

- **Экземпляр 1** имеет адрес <https://demo03-web2025.alteroffice.ru>

Параметры, которые потребуются для копирования и восстановления:

Параметр	Описание	Пример
<b>PATH_SERVER</b>	Путь к директории на хосте, где установлена система АльтерОфис Веб	/opt/alterofficeweb/demo 03
<b>PATH_BACKUP</b>	Путь к директории для хранения (временного) резервных копий.	/home/alterofficeweb/backup
<b>APP_CONTAINER_NAME</b>	Имя контейнера основного веб-приложения.	prod-app-1
<b>FRONT_CONTAINER_NAME</b>	Имя контейнера, обеспечивающего балансировку нагрузки, кэширование статического содержимого, перенаправления запросов в другие контейнеры.	prod-front-1
<b>CRON_CONTAINER_NAME</b>	Имя контейнера, обеспечивающего выполнение фоновых задач.	prod-cron-1
<b>EDITORS_CONTAINER_NAME</b>	Имя контейнера, обеспечивающего совместное редактирование документов.	prod-editors-1
<b>DB_CONTAINER_NAME</b>	Имя контейнера, обеспечивающего хранение служебной информации, пользователей, метаданных, настроек.	prod-db-1

## 5.1. Резервное копирование АльтерОфис Веб

Рекомендуется регулярно создавать резервные копии системы: данных, конфигураций, настроек.

Ниже будет рассмотрено несколько вариантов резервного копирования.

:::tip[Предварительные требования] Перед выполнением команд Docker необходимо:

- добавить текущего пользователя в группу docker или
- использовать sudo для запуска команд с правами суперпользователя. :::

---

### 5.1.1. Полное копирование системы

Создаётся полная копия всей инфраструктуры, включая:

- ядро системы (файлы приложения);
- база данных (PostgreSQL);
- конфигурационные файлы;
- пользовательские данные;
- хранилище MinIO (если используется).

#### Важно

Восстановление возможно только в рамках одной версии продукта (например, резервная копия версии 2025.0.0.8 несовместима с версией 2025.0.0.9).

**Шаг 1.** Перейдите в каталог, где установлена система.

`cd <PATH_SERVER>`

Пример:

`cd /opt/alterofficeweb/demo03`

**Шаг 2.** Остановите контейнер с онлайн-редакторами.

Требуется остановить контейнер **editors**.

`sudo docker stop <EDITORS_CONTAINER_NAME>`

Пример:

`sudo docker stop prod-editors-1`

**Шаг 3.** Остановите контейнеры АльтерОфис Веб (контейнер с базой данных продолжает работать).

Требуется остановить контейнеры **app**, **cron**, **front**.

`sudo docker stop <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>`

Пример:

```
sudo docker stop prod-app-1 prod-cron-1 prod-front-1
```

**Шаг 4.** Создайте директорию для резервного копирования

```
mkdir -p <PATH_BACKUP>
```

Пример:

```
mkdir -p /home/alterofficeweb/backup
```

Назначьте необходимые права доступа к директории.

**Шаг 5.** Создайте архив с данными резервного копирования:

```
sudo tar -czf <PATH_BACKUP>/app_$(date +%Y_%m_%d_%H_%M).tar.gz <PATH_SERVER>
```

Замените:

- <PATH\_BACKUP> на путь к директории резервного копирования;
- <PATH\_SERVER> на путь, где установлена система.

Пример:

```
sudo tar -czf /home/alterofficeweb/backup/app_$(date +%Y_%m_%d_%H_%M).tar.gz /opt/alterofficeweb/demo03
```

**Шаг 6.** Создайте резервный дамп ролей баз данных:

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dumpall -U "$POSTGRES_USER" --roles-only | gzip > <PATH_BACKUP>/roles_$(date +%Y_%m_%d_%H_%M).sql.gz
```

Пример:

```
source .env && sudo docker exec prod-db-1 pg_dumpall -U "$POSTGRES_USER" --roles-only | gzip > /home/alterofficeweb/backup/roles_$(date +%Y_%m_%d_%H_%M).sql.gz
```

**Шаг 7.** Создайте резервный дамп базы данных:

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dump -U "$POSTGRES_USER" "$POSTGRES_DB" | gzip > <PATH_BACKUP>/db_$(date +%Y_%m_%d_%H_%M).sql.gz
```

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dump -U "$POSTGRES_USER" "$POSTGRES_DB" | gzip > /home/alterofficeweb/backup/db_$(date +%Y_%m_%d_%H_%M).sql.gz
```

Резервные копии созданы и находятся в директории /home/alterofficeweb/backup.

Рекомендуется хранить резервные копии отдельно от данных системы (на отдельном сервере или использовать специализированные системы хранения данных).

**Шаг 8.** Запустите ранее остановленные контейнеры АльтерОфис Веб

Требуется запустить контейнеры **app, cron, front**.

```
sudo docker start <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-app-1 prod-cron-1 prod-front-1
```

**Шаг 9.** Запустите ранее остановленный контейнер с онлайн-редакторами.

Требуется запустить контейнер **editors**.

```
sudo docker start <EDITORS_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-editors-1
```

На данном этапе создание полной резервной копии системы завершено. Храните резервные копии отдельно от серверов, на которых установлена система.

Периодически проверяйте корректность создания резервных копий, путем восстановления системы из резервных копий на тестовом стенде.

---

### 5.1.2. Резервное копирование пользовательских данных

Сохраняются только пользовательские данные, конфигурационные файлы и база данных.

Преимущество

Позволяет выполнять восстановление на разных версиях продукта.

**Шаг 1.** Перейдите в каталог, где установлена система.

```
cd <PATH_SERVER>
```

Пример:

```
cd /opt/alterofficeweb/demo03
```

**Шаг 2.** Остановите контейнер с онлайн-редакторами.

Требуется остановить контейнер **editors**.

```
sudo docker stop <EDITORS_CONTAINER_NAME>
```

Пример:

```
sudo docker stop prod-editors-1
```

**Шаг 3.** Остановите контейнеры АльтерОфис Веб (контейнер с базой данных продолжает работать).

Требуется остановить контейнеры **app, cron, front**.

```
sudo docker stop <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>
```

Пример:

```
sudo docker stop prod-app-1 prod-cron-1 prod-front-1
```

**Шаг 4.** Создайте директорию для резервного копирования

```
mkdir -p <PATH_BACKUP>
```

Пример:

```
mkdir -p /home/alterofficeweb/backup
```

Назначьте необходимые права доступа к директории.

**Шаг 5.** Создайте архив с данными резервного копирования:

```
sudo tar -czf <PATH_BACKUP>/user_data_$(date +%Y_%m_%d_%H_%M).tar.gz <PATH_SERVER>/html/data <PATH_SERVER>/html/config
```

Замените:

- <PATH\_BACKUP> на путь к директории резервного копирования;
- <PATH\_SERVER> на путь, где установлена система.

Пример:

```
sudo tar -czf /home/alterofficeweb/backup/user_data_$(date +%Y_%m_%d_%H_%M).tar.gz /opt/alterofficeweb/demo03/html/data /opt/alterofficeweb/demo03/html/config
```

**Шаг 6.** Создайте резервный дамп ролей баз данных:

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dumpall -U "$POSTGRES_USER" --roles-only | gzip > <PATH_BACKUP>/roles_$(date +%Y_%m_%d_%H_%M).sql.gz
```

Пример:

```
source .env && sudo docker exec prod-db-1 pg_dumpall -U "$POSTGRES_USER" --roles-only | gzip > /home/alterofficeweb/backup/roles_$(date +%Y_%m_%d_%H_%M).sql.gz
```

**Шаг 7.** Создайте резервный дамп базы данных:

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dump -U "$POSTGRES_USER" "$POSTGRES_DB" | gzip > <PATH_BACKUP>/db_$(date +%Y_%m_%d_%H_%M).sql.gz
```

Пример:

```
source .env && sudo docker exec <DB_CONTAINER_NAME> pg_dump -U "$POSTGRES_USER" "$POSTGRES_DB" | gzip > /home/alterofficeweb/backup/db_$(date +%Y_%m_%d_%H_%M).sql.gz
```

Резервные копии созданы и находятся в директории /home/alterofficeweb/backup.

Рекомендуется хранить резервные копии отдельно от данных системы (на отдельном сервере или использовать специализированные системы хранения данных).

#### **Шаг 8.** Запустите ранее остановленные контейнеры АльтерОфис Веб

Требуется запустить контейнеры **app, cron, front**.

```
sudo docker start <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-app-1 prod-cron-1 prod-front-1
```

#### **Шаг 9.** Запустите контейнер с онлайн-редакторами

```
sudo docker start <EDITORS_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-editors-1
```

На этом резервное копирование пользовательских данных завершено.

---

## 5.2. Восстановление АльтерОфис Веб из резервных копий

#### **Шаг 1.** Перейдите в каталог, где установлена система.

```
cd <PATH_SERVER>
```

Замените <PATH\_SERVER> на путь, где установлена система.

Пример:

```
cd /opt/alterofficeweb/demo03
```

#### **Шаг 2.** Остановите контейнер с онлайн-редакторами.

```
sudo docker stop <EDITORS_CONTAINER_NAME>
```

Пример:

```
sudo docker stop prod-editors-1
```

#### **Шаг 3.** Остановите контейнеры с АльтерОфис Веб.

```
sudo docker stop <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>
```

Пример:

```
sudo docker stop prod-app-1 prod-cron-1 prod-front-1
```

#### **Шаг 4.** Распакуйте архив с резервной копией

- Для полного восстановления системы:

```
sudo tar -xzf <PATH_BACKUP>/app_<*****>.tar.gz -C /
```

- Для восстановления пользовательских данных:

```
sudo tar -xzf <PATH_BACKUP>/user_data_<*****>.tar.gz -C /
```

Замените:

- <PATH\_BACKUP> на путь к резервной копии.
- <\*\*\*\*\*> на дату и время создания резервной копии.

Пример:

- Полное восстановление системы:

```
sudo tar -xzf /home/alterofficeweb/backup/app_2025_11_25_20_44.tar.gz -C /
```

- Восстановление пользовательских данных:

```
sudo tar -xzf /home/alterofficeweb/backup/user_data_2025_11_25_20_44.tar.gz -C /
```

**Шаг 5.** Удалите текущую базу данных

```
source .env && sudo docker exec <DB_CONTAINER_NAME> psql -U "$POSTGRES_USER" -d template1 -c "DROP DATABASE \"$POSTGRES_DB\";"
```

Пример:

```
source .env && sudo docker exec prod-db-1 psql -U "$POSTGRES_USER" -d template1 -c "DROP DATABASE \"$POSTGRES_DB\";"
```

**Шаг 6.** Создайте новую базу данных

```
source .env && sudo docker exec <DB_CONTAINER_NAME> psql -U "$POSTGRES_USER" -d template1 -c "CREATE DATABASE \"$POSTGRES_DB\";"
```

Пример:

```
source .env && sudo docker exec prod-db-1 psql -U "$POSTGRES_USER" -d template1 -c "CREATE DATABASE \"$POSTGRES_DB\";"
```

**Шаг 7.** Восстановите роли базы данных из резервной копии

```
source .env && gunzip -c <PATH_BACKUP>/roles_<*****>.sql.gz | sudo docker exec -i <DB_CONTAINER_NAME> psql -U "$POSTGRES_USER" -d postgres
```

Замените:

- <PATH\_BACKUP> на путь к резервной копии.
- <DB\_CONTAINER\_NAME> на имя контейнера, содержащего базу данных системы.
- <\*\*\*\*\*> на дату и время создания резервной копии.

Пример:

```
source .env && gunzip -c /home/alterofficeweb/backup/roles_2025_11_25_20_44.sql.gz | sudo docker exec -i prod-db-1 psql -U "$POSTGRES_USER" -d postgres
```

**Шаг 8.** Восстановите данные из резервной копии

```
source .env && gunzip -c <PATH_BACKUP>/db_<*****>.sql.gz | sudo docker exec -i <DB_CONTAINER_NAME> psql -U "$POSTGRES_USER" -d "$POSTGRES_DB"
```

Пример:

```
source .env && gunzip -c /home/alterofficeweb/backup/db_2025_11_25_20_44.sql.gz | sudo docker exec -i prod-db-1 psql -U "$POSTGRES_USER" -d "$POSTGRES_DB"
```

**Шаг 9.** Запустите контейнеры с АльтерОфис Веб.

```
sudo docker start <APP_CONTAINER_NAME> <CRON_CONTAINER_NAME> <FRONT_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-app-1 prod-cron-1 prod-front-1
```

**Шаг 10.** Запустите контейнеры с онлайн-редакторами АльтерОфис Веб

```
sudo docker start <EDITORS_CONTAINER_NAME>
```

Пример:

```
sudo docker start prod-editors-1
```

На данном этапе восстановление системы завершено.

---

## 6. Процедура обновления сертификата и резервное копирование сертификатов

### 6.1. Создание резервной копии текущих сертификатов

**Шаг 1.** На хостовой машине перейдите в директорию, где лежат сертификаты:

```
cd <PATH_SSL>
```

Замените:

- <PATH\_SSL> на путь к директории с SSL сертификатами.

Пример:

```
cd /etc/ssl/alteroffice
```

**Шаг 2.** Создайте архив текущих сертификатов:

```
sudo tar -czf <PATH_BACKUP>/alteroffice_ssl_backup_$(date +%Y_%m_%d_%H_%M).tar.gz -C <PATH_SSL> .
```

Замените:

- <PATH\_BACKUP> на путь к директории резервного копирования;
- <PATH\_SSL> на путь к директории с SSL сертификатами.

Пример:

```
sudo tar -czf /home/alterofficeweb/backup/alteroffice_ssl_backup_$(date +%Y_%m_%d_%H_%M).tar.gz -C /etc/ssl/alteroffice .
```

---

### 6.2. Установка новых сертификатов

**Шаг 1.** Скопируйте новые сертификаты

```
sudo cp server.key <PATH_BACKUP>/  
sudo cp server.crt <PATH_BACKUP>/
```

Пример:

```
sudo cp server.key /etc/ssl/alteroffice/  
sudo cp server.crt /etc/ssl/alteroffice/
```

**Шаг 2.** Установите правильные права доступа

```
sudo chown root:root <PATH_BACKUP>/*  
sudo chmod 644 <PATH_BACKUP>/server.key  
sudo chmod 600 <PATH_BACKUP>/server.crt
```

Пример:

```
sudo chown root:root /etc/ssl/alteroffice/*
sudo chmod 644 /etc/ssl/alteroffice/server.key
sudo chmod 600 /etc/ssl/alteroffice/server.crt
```

**Шаг 3.** Перезапустите сервис **front** для применения сертификата

```
cd <PATH_SERVER>
sudo docker restart <FRONT_CONTAINER_NAME>
```

Пример:

```
cd /opt/alterofficeweb/demo03
sudo docker restart prod-front-1
```

**Шаг 4.** Проверьте валидность сертификата (вместо *stage.my.local* укажите адрес своего сервера)

```
echo | openssl s_client -connect stage.my.local:443 2>/dev/null | openssl x509 -noout -dates -subject
```

---

### 6.3. Восстановление старых сертификатов (в случае ошибки)

**Шаг 1.** Перейдите в директорию:

```
cd <PATH_SSL>
```

Замените:

- <PATH\_SSL> на путь к директории с SSL сертификатами.

Пример:

```
cd /etc/ssl/alteroffice
```

**Шаг 2.** Удалите новые файлы:

```
sudo rm server.key server.crt
```

**Шаг 3.** Распакуйте резервную копию (укажите имя вашего архива):

```
sudo tar -xzf <PATH_BACKUP>/alteroffice_ssl_backup_<*****>.tar.gz -C <PATH_SSL>
```

Замените:

- <PATH\_BACKUP> на путь к директории резервного копирования;
- <PATH\_SSL> на путь к директории с SSL сертификатами.
- <\*\*\*\*\*> на дату и время создания резервной копии.

Пример:

```
sudo tar -xzf /home/alterofficeweb/backup/alteroffice_ssl_backup_2025_11_25_20_4
4.tar.gz -C /etc/ssl/alteroffice
```

**Шаг 4.** Убедитесь, что права восстановлены:

```
sudo chmod 644 server.key
sudo chmod 600 server.crt
```

**Шаг 5.** Перезапустите **front**

```
cd <PATH_SERVER>
sudo docker restart <FRONT_CONTAINER_NAME>
```

Пример:

```
cd /opt/alterofficeweb/demo03
sudo docker restart prod-front-1
```

---

## 7. Полная деинсталляция

В зависимости от того, как был установлен Docker Compose, запуск будет либо docker-compose, либо docker compose.

**Шаг 1.** Перейти в директорию откуда запускалась инсталляция

**Шаг 2.** Выполнить поочередно:

```
docker-compose -f compose.yaml --project-name prod down -v
docker-compose -f compose-elasticsearch.yaml --project-name prod-elk down -v
docker network rm alterofficeweb
```

**Шаг 3.** Удалить директории и файлы конфигурации:

- “Директория контента” - посмотреть в `installation_backup_***.txt`
- Удалить, если есть, директорию **db**.
- Удалить файлы конфигурации `.env`, все `compose` файлы.

---

## 8. Обновление системы АльтерОфис Веб

### Внимание

Данная инструкция подходит для версии инсталлятора 0.0.9.

**Шаг 1.** Останавливаем приложение.

```
docker-compose -f compose.yaml --project-name prod stop app cron front
```

**Шаг 2.** Делаем полную резервную копию данных, конфигурации, базы данных.

**Шаг 3.** Скачиваем новую версию из репозитория <https://repo.alteroffice.ru/web/> любым удобным способом:

- Архив с установщиком alterofficeweb-installer\_\*.zip.
- Набор Docker-образов alterofficeweb-bundle\_\*.tar.

**Шаг 4.** Распакуйте архив с установщиком в текущую директорию, с заменой существующих файлов:

```
sudo unzip -o alterofficeweb-installer_*.zip
```

**Шаг 5.** Загрузите образы из bundle:

```
sudo docker load -i alterofficeweb-bundle_*.tar
```

**Шаг 6.** Перезапускаем приложение

```
docker-compose -f compose-elasticsearch.yaml --project-name prod-elk up -d
docker-compose -f compose.yaml --project-name prod up -d
```

### Примечание

Если в процессе обновления возникнут ошибки, АльтерОфис Веб автоматически перейдет в режим обслуживания.

## 9. Подключение S3-совместимого хранилища как основного хранилища пользовательских данных

Данный раздел описывает, как подключить существующее S3-совместимое хранилище (например, MinIO или другое) в качестве основного хранилища данных системы.

В качестве примера приведена настройка для **MinIO**, который по умолчанию уже развернут.

### 9.1. Настройка основного хранилища Minio на уже развернутой системе

#### Внимание

Настройка основного S3-хранилища объектов сделает все существующие локальные файлы недоступными.

#### Шаг 1. Подготовка конфигурационного файла

Создайте файл конфигурации для системы, например, **storage.config.php**, который будет содержать параметры подключения к вашему S3-совместимому хранилищу.

Ниже приведен пример для **MinIO**:

```
<?php
$CONFIG = array ('objectstore' => array(
    'class' => '\OC\Files\ObjectStore\S3',
    'arguments' => array(
        'bucket' => 'имя_вашего_бакета',
        'key' => 'ваш_access_key',
        'secret' => 'ваш_secret_key',
        'hostname' => 'адрес_хранилища',
        'port' => порт_подключения,
        'use_ssl' => false, // или true, если используется SSL
        'use_path_style' => true, // рекомендуется для MinIO и аналогичных решений
    ),
),
);
?>
```

#### Важно

Параметры hostname, port, key, secret, bucket должны соответствовать настройкам вашего хранилища. Для других решений могут потребоваться дополнительные параметры или отличаться названия.

#### Шаг 2. Размещение файла конфигурации

Скопируйте подготовленный файл в директорию конфигурации вашей системы:

```
sudo cp storage.config.php /var/www/alteroffice/html/config/  
sudo chown www-data:www-data /var/www/alteroffice/html/config/storage.config.php
```

### Шаг 3. Проверка конфигурации:

Чтобы убедиться в правильности настройки, выполните команду внутри контейнера:

```
docker exec -it <имя_контейнера> /bin/bash  
.occ config:system:get objectstore
```

Вы должны увидеть параметры, соответствующие вашей конфигурации.

#### Внимание

После добавления файла конфигурации изменения вступают в силу автоматически. Существующие файлы останутся в локальной папке.

## 9.2. Настройка MinIO в качестве основного хранилища при развертывании с нуля

#### Важно

Настройка Minio и его развертывание осуществляется вне рамок этой инструкции. Предполагается, что выбранное хранилище уже развернуто и доступно.

### Шаг 1. Настройка окружения контейнера приложения.

В файле конфигурации контейнера (**compose.yaml** или аналогичном) для сервиса **app** необходимо указать переменные окружения, содержащие параметры подключения к хранилищу:

- OBJECTSTORE\_S3\_HOST=\$MINIO\_HOST
- OBJECTSTORE\_S3\_PORT=\$MINIO\_PORT
- OBJECTSTORE\_S3\_BUCKET=\$MINIO\_BUCKET
- OBJECTSTORE\_S3\_SSL=\$MINIO\_SSL
- OBJECTSTORE\_S3\_KEY=\$MINIO\_KEY
- OBJECTSTORE\_S3\_SECRET=\$MINIO\_SECRET
- OBJECTSTORE\_S3\_AUTOCREATE=\$MINIO\_AUTOCREATE
- OBJECTSTORE\_S3\_USEPATH\_STYLE=\$MINIO\_USEPATH\_STYLE

Пример:

- OBJECTSTORE\_S3\_HOST=minio
- OBJECTSTORE\_S3\_PORT=9000
- OBJECTSTORE\_S3\_BUCKET=aow-external
- OBJECTSTORE\_S3\_SSL=false
- OBJECTSTORE\_S3\_KEY=alterofficeweb
- OBJECTSTORE\_S3\_SECRET=alterofficeweb
- OBJECTSTORE\_S3\_AUTOCREATE=true
- OBJECTSTORE\_S3\_USEPATH\_STYLE=true

Параметры, которые необходимы для настройки:

Параметр	Описание	Пример
<b>OBJECTSTORE_S3_HOST</b>	Хост (URL-эндпоинт) Minio	minio
<b>OBJECTSTORE_S3_PORT</b>	Порт для подключения к хранилищу	9000
<b>OBJECTSTORE_S3_BUCKET</b>	Имя бакета, который будет использован для хранения данных	aow-external
<b>OBJECTSTORE_S3_SSL</b>	Использовать ли SSL/TLS для соединения. Указывайте <code>false</code> только для тестирования с self-signed сертификатами. Значение по умолчанию: <code>true</code>	<code>false</code>
<b>OBJECTSTORE_S3_KEY</b>	Ключ доступа (Access Key) для вашего объектного хранилища. Совпадает с указанным <code>MINIO_ROOT_USER</code>	alterofficeweb
<b>OBJECTSTORE_S3_SECRET</b>	Секретный ключ (Secret Key) для вашего объектного хранилища. Совпадает с указанным <code>MINIO_ROOT_PASSWORD</code>	alterofficeweb
<b>OBJECTSTORE_S3_AUTOCREATE</b>	Создать бакет, если он не существует. Значение по умолчанию: <code>true</code>	<code>true</code>
<b>OBJECTSTORE_S3_USEPATH_STYLE</b>	Использовать path-style URL для доступа к бакетам. Установите в <code>true</code> для совместимости с некоторыми провайдерами, такими как MinIO или старые версии Серф. Значение по умолчанию: <code>false</code>	<code>true</code>

Эти переменные позволяют системе динамически определить параметры подключения к вашему хранилищу.

Затем эти переменные можно использовать в конфигурационных файлах системы или передавать как переменные окружения при запуске контейнеров.

**Шаг 2.** Запуск скрипта установки.

Продолжить с **Шага №2** инструкции в разделе **4.2**.

В такой конфигурации все пользовательские данные будут созданы в указанном S3 хранилище.

**Примечание**

В случае потери соединения с хранилищем или неправильной настройки, система может отображать ошибку, например: “Внутренняя ошибка сервера. Запрос не может быть обработан сервером.”