

**ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»**

**СЕРВЕР СОВМЕСТНОГО РЕДАКТИРОВАНИЯ**

**3.2**

**АРХИТЕКТУРА**

**Версия 1**

**На 15 листах**

**Дата публикации: 17.12.2024**

**Москва  
2024**

# МойОфис

Все упомянутые в этом документе названия продуктов, логотипы, торговые марки и товарные знаки принадлежат их владельцам.

Товарные знаки «МойОфис», «MyOffice» и «Squadus» принадлежат ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ».

Ни при каких обстоятельствах нельзя истолковывать любое содержимое настоящего документа как прямое или косвенное предоставление лицензии или права на использование товарных знаков, логотипов или знаков обслуживания, приведенных в нем. Любое несанкционированное использование этих товарных знаков, логотипов или знаков обслуживания без письменного разрешения их правообладателя строго запрещено.

## СОДЕРЖАНИЕ

1	Общие сведения .....	5
1.1	Назначение .....	5
1.2	Общая компонентная схема .....	5
2	Описание архитектуры .....	6
2.1	Архитектурная схема Сервера совместного редактирования .....	6
2.1.1	Описание архитектуры Сервера совместного редактирования .....	7
3	Типовые схемы установки .....	8
3.1	Конфигурация без отказоустойчивости .....	8
3.2	Кластерная отказоустойчивая конфигурация .....	9
3.3	Требования для кластера с профилем более 2000 пользователей .....	10
3.4	Типовая схема масштабирования .....	10
3.5	Системные учетные записи .....	10
	Приложение А. Описание ролей при расчете аппаратных требований .....	11
	Приложение Б. Пример файла inventory (установка standalone) .....	12
	Приложение В. Пример файла inventory (кластерная установка) .....	14

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ, ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ

В настоящем документе используются следующие сокращения с соответствующими расшифровками (табл. 1).

Таблица 1 — Сокращения и обозначения

Сокращение, термин	Расшифровка и определение
API	Application Programming Interface, интерфейс программирования приложений
Auth SSO	Single Sign-On, процесс аутентификации, позволяющий пользователю один раз войти в систему с одним набором учетных данных для доступа к нескольким приложениям или службам
AMQP	Advanced Message Queuing Protocol, открытый протокол прикладного уровня для передачи сообщений между компонентами систем
REST API	Архитектурный стиль взаимодействия компонентов распределенного приложения в сети
SPA	Single Page Application, веб-приложение или веб-сайт, использующий единственный HTML-документ как оболочку для всех веб-страниц и организующий взаимодействие с пользователем через динамически подгружаемые HTML, CSS, JavaScript, обычно посредством AJAX
SIEM	Security information and event management, система (или технология) управления информацией и событиями безопасности
Inventory	Файл, содержащий набор управляемых хостов для автоматизации установки и управления конфигурацией для сервиса Ansible
IOPS	Input/Output Operations Per Second, количество операций ввода/вывода
UI	User interface, интерфейс пользователя
ОС	Операционная система
СУБД	Система управления базами данных, комплекс программно-языковых средств, позволяющих создать базы данных и управлять данными
Тенант	Логический объект, включающий в себя совокупность вычислительных ресурсов, репозиторий и пользователей
ПО	Программное обеспечение

# МойОфис

## 1 ОБЩИЕ СВЕДЕНИЯ

### 1.1 Назначение

В настоящем документе описана архитектура продукта и взаимодействие сервисов Сервера совместного редактирования (ССР).

### 1.2 Общая компонентная схема

На рисунке 1 представлена общая схема взаимодействия компонентов продукта.

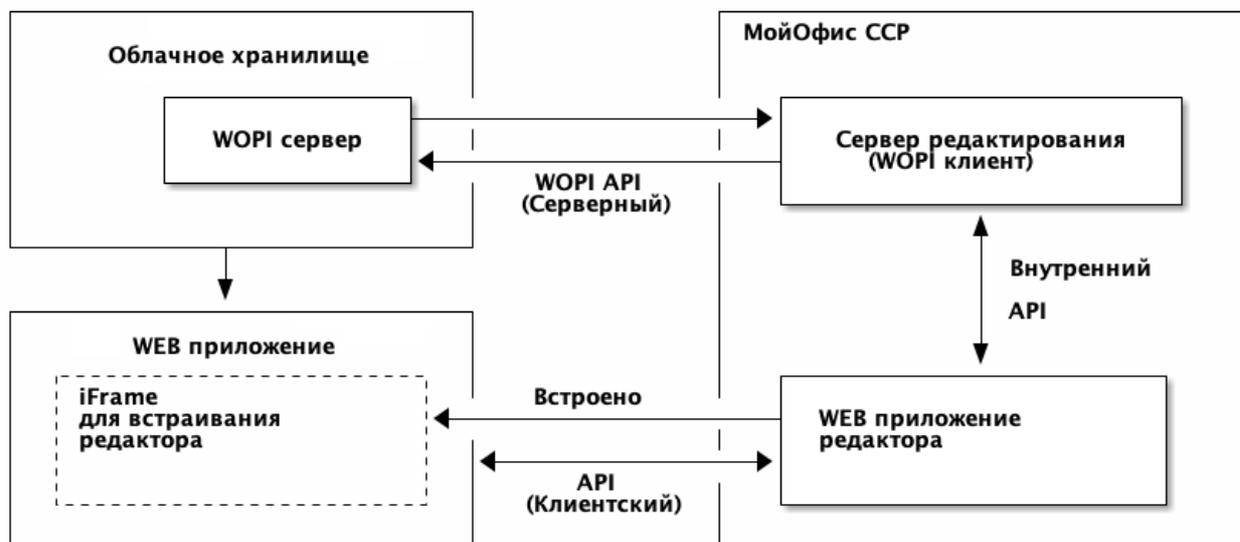


Рисунок 1 — Общая компонентная схема

## 2 ОПИСАНИЕ АРХИТЕКТУРЫ

### 2.1 Архитектурная схема Сервера совместного редактирования

На рисунке 2 представлена схема взаимодействия подсистем Сервера совместного редактирования. Прямоугольниками выделены кластерные решения для сервисов `etcd`, `redis`, `rabbitmq`.

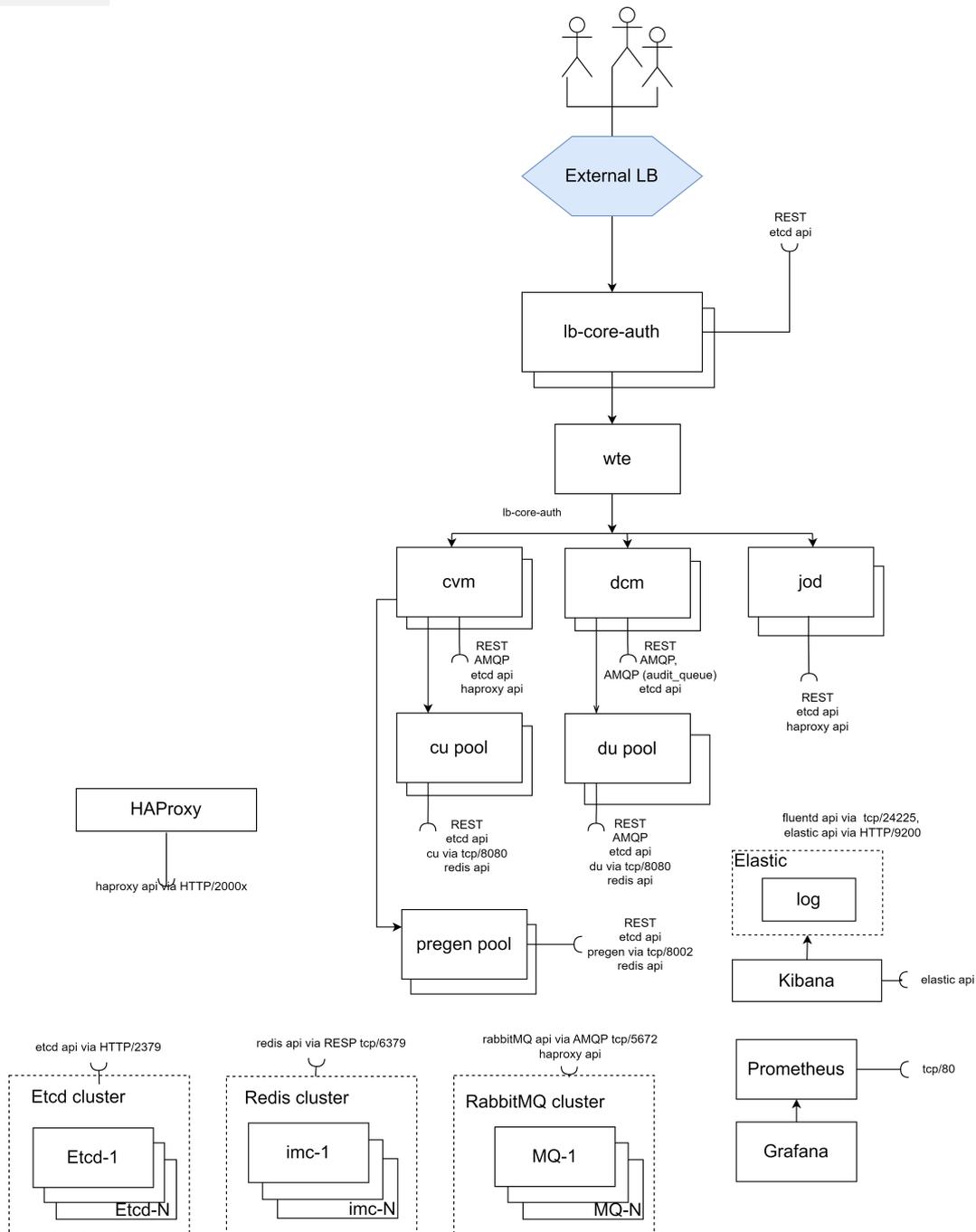


Рисунок 2 — Архитектурная схема Сервера совместного редактирования

## 2.1.1 Описание архитектуры Сервера совместного редактирования

Описание подсистем и сервисов Сервера совместного редактирования приведено в таблице 2.

Таблица 2 — Описание подсистем и сервисов Сервера совместного редактирования

Наименование подсистемы или сервиса	Описание
Lb-core-auth (nginx-wfe)	Расширенный веб-сервер NGINX с поддержкой Lua. Отвечает за авторизацию, доступность API, балансировку
HAProxy	Балансировщик нагрузки между внутренними сервисами
CVM	Conversion Manager, сервис управления конвертированием файлов. При конвертации документов использует пул CU юнитов, Pregen или Jod
CU	Conversion Unit, экземпляр процесса конвертора различных форматов файлов (офисных или изображений)
Pregen	Сервис конвертации документов в форматы PDF и JSON
DCM	Document Collaboration Manager, сервис управления редактированием документов. При редактировании документов использует пул DU юнитов
DU	Document Unit, экземпляр процесса редактирования и коллаборации документов
JOD	Java OpenDocument Converter, сервис конвертации офисных файлов устаревших форматов (например Microsoft Office 1997) в современные форматы или PDF
Elastic	Elastic (Opendistro Elastic Search), поисковый и аналитический сервис. Отвечает за сбор, обработку и хранение логов всех сервисов
Kibana	Kibana, инструмент визуализации и изучения данных, анализ логов приложения, использует данные из Elastic
EtcD	Хранилище типа «ключ-значение». Отвечает за хранение свойств (настроек) всех сервисов
Redis	поSQL in-мемори хранилище, типа «ключ-значение». Отвечает за хранение кешированной информации при работе с облаком (файлы, профили пользователей, токены)
RabbitMQ	Брокер сообщений на основе протокола AMQP. Отвечает за отправку сообщений между сервисами, например за отправку уведомлений, событий аудита
Prometheus	Инструмент сбора метрик со всех сервисов
Grafana	Инструмент визуализации метрик приложения. Получает метрики из Prometheus
WTE	Web text editors (Веб-клиент офисного приложения). Отвечает за редактирование и чтение файлов

## 3 ТИПОВЫЕ СХЕМЫ УСТАНОВКИ

### 3.1 Конфигурация без отказоустойчивости

Конфигурация без отказоустойчивости может использоваться при условии предоставления отказоустойчивости на уровне виртуализации.

Данная конфигурация характеризуется тем, что все серверные роли развертываются на единственном сервере. В такой конфигурации роли устанавливаются:

- на несколько виртуальных серверов с объединением ролей;
- на несколько виртуальных серверов, где одному серверу соответствует одна роль.

Установка такого типа не требует использования подсистемы балансировки.

Пример расчета аппаратных требований для установки на двух виртуальных серверах приведен в таблице 3.

В данном разделе приведены требования для развертывания системы без отказоустойчивости со следующим максимальным профилем эксплуатации:

- всего пользователей — 999;
- количество одновременно активных пользователей — 400;
- количество документов, редактируемых одновременно — 100.



Существует ограничение на количество пользователей  $\leq 1000$  из-за невозможности масштабирования конфигурации без отказоустойчивости (standalone). Требования идентичны для систем с общим количеством пользователей от 1 до 999.

Таблица 3 — Аппаратные требования для конфигурации без отказоустойчивости

Роли серверов	CPU, vCPU	RAM, Гбайт	SSD, Гбайт
operator*	1	4	50
Все роли Сервера совместного редактирования	8	20	100

\* — сервер с ролью operator рекомендуется размещать на отдельном виртуальном сервере. После установки сервер с ролью operator не используется и может потребоваться для переустановки системы или ее сервисов.

Примеры файла inventory для Сервера совместного редактирования представлены в приложении Б.

## 3.2 Кластерная отказоустойчивая конфигурация

В кластерной отказоустойчивой конфигурации каждая критическая роль реплицируется на разных виртуальных серверах. Разные роли могут быть объединены на одном виртуальном сервере. Архитектурных ограничений по объединению ролей нет. Виртуальные серверы разносятся по разным физическим серверам или гипервизорам.

Если один из серверов роли прекратил свою работу, общая распределенная роль продолжит работу сервиса на других серверах, и система сохранит свою работоспособность в полном объеме. Если все сервера роли прекратят работу, то система потеряет часть функциональности или станет полностью недоступна.

В данном разделе приведены минимальные требования для развертывания системы в режиме кластера со следующим профилем эксплуатации:

- всего пользователей — 2000;
- количество одновременно активных пользователей — 1000;
- количество документов, редактируемых одновременно — 90.

Пример расчета аппаратных требований для отказоустойчивой установки приведен в таблице 4.

Таблица 4 — Аппаратные требования для кластерной отказоустойчивой конфигурации

Наименование роли*	Количество серверов	CPU, vCPU	RAM, Гбайт	SSD, Гбайт	HDD, Гбайт
operator	1	1	4	0	50
LB	2	2	4	50	0
core	2	12	24	100	0
infra	1	2	12	100	0
mq+imc+etcd	3	6	16	50	0

\* — описания ролей представлены в приложении А

При распределении ролей для кластерной отказоустойчивой конфигурации необходимо учитывать следующие требования:

- сервер `operator` не рекомендуется устанавливать на одном виртуальном сервере с приложением, но он может быть выключен после установки;
- для кластерных ролей `etcd` / `mq` / `imc` необходимо использовать минимум 3 узла, для `etcd` рекомендуется использовать 5 узлов;
- роли `cvm/cu-pool` и `dcm/du-pool` в такой конфигурации объединены;
- роль `PGS-LOG` не является критической;
- роль `LB` — внешний балансировщик.

Примеры файла `inventory` для Сервера совместного редактирования представлены в приложении В.

### 3.3 Требования для кластера с профилем более 2000 пользователей

Для кластерной установки, на которой планируется работа более 2000 пользователей, необходимо обратиться к вендору для расчета размеров серверных ресурсов и получения рекомендаций по объединению ролей.

### 3.4 Типовая схема масштабирования

Полноценное масштабирование возможно использовать только для кластерной отказоустойчивой конфигурации.

Для standalone конфигурации возможно использование вертикального масштабирования с учетом ограничения Docker и других системных сервисов.

Переход от standalone конфигурации к кластерной выполняется при полной переустановке продукта. На сервере хранения данных переход обеспечивается с помощью предварительного резервирования баз данных.

Для в первую очередь следует масштабировать узлы кластера с ролями:

- `dcm`, `du-pool` и `cu-pool` (влияет на количество одновременно открытых документов);
- `cvm` и `pregen` (влияет на количество конвертаций, скорость загрузки, скачивания, печати документов).

### 3.5 Системные учетные записи

В таблице 5 и 6 представлены системные учетные записи, необходимые для связи сервисов системы.

Таблица 5 — Системные учетные записи

Имя учетной записи	Имя сервиса	Описание
couser	Etcd_browser	UI конфигурации Etcd
couser	Openresty	Пользователь CO Manage API
admin	Grafana	Пользователь Grafana UI
admin	Elasticsearch	Пользователь Elasticsearch/Kibana UI
kibana	Elasticsearch	Пользователь для связи Kibana и Elasticsearch
couser	RabbitMQ	Пользователь RabbitMQ Management UI и подключения сервисов
root	RabbitMQ	Пользователь RabbitMQ для управления очередями и конфигурации во время деплоя

## ПРИЛОЖЕНИЕ А

### Описание ролей при расчете аппаратных требований

Таблица А.1 — Роли для установки standalone

Наименование	Описание
CO	Содержит службы SSO, веб-редакторов, collaboration API

Таблица А.2 — Роли для кластерной установки Системы редактирования и совместной работы

Таблица А.2 — Роли для кластерной установки Сервера совместного редактирования

Наименование	Описание
lb-core-auth	Сервер балансировки нагрузки Системы редактирования и совместной работы
infra	Сервер, объединяющий инфраструктурные роли сбора логов и мониторинга Системы редактирования и совместной работы. Может содержать роль chatbot
pregen	Сервер генерации превью и индексных документов
etcd	Подсистема конфигурации с использованием Etcd
core-cvm	Сервис управления импортом, экспортом и индексированием документов
cu-pool	Пул контейнеров с конвертерами документов
core-dcm	Сервер управления редактированием, коллаборации и документного API
du pool	Пул контейнеров с модулями редактирования документов в режиме коллаборации
imc	Сервер кэширования сессий и хранения промежуточных результатов в памяти
mq	Сервер очереди сообщений и подписок
core	При сокращенном составе ролей — совмещенные роли *-core-* для Системы редактирования и совместной работы

Таблица А.3 — Технические роли

Наименование	Описание
operator	Технологическая роль. Рабочее место, с которого производится установка всех компонентов
LB	Сервер балансировки нагрузки для всех компонентов (используется только при кластерной установке)

## ПРИЛОЖЕНИЕ Б

### Пример файла inventory Системы редактирования и совместной работы (установка standalone)

```
all:
  children:
    co:
      children:
        co_chatbot:
          hosts:
            co-infra-1.installation.example.net:
        co_etcd:
          hosts:
            co-infra-1.installation.example.net:
        co_mq:
          hosts:
            co-infra-1.installation.example.net:
        co_cvm:
          hosts:
            co-infra-1.installation.example.net:
        co_cu:
          hosts:
            co-infra-1.installation.example.net:
        co_dcm:
          hosts:
            co-infra-1.installation.example.net:
        co_du:
          hosts:
            co-infra-1.installation.example.net:
        co_fm:
          hosts:
            co-infra-1.installation.example.net:
        co_jod:
          hosts:
            co-infra-1.installation.example.net:
        co_nm:
          hosts:
            co-infra-1.installation.example.net:
        co_pregen:
          hosts:
            co-infra-1.installation.example.net:
        co_imc:
          hosts:
            co-infra-1.installation.example.net:
        co_lb_core_auth:
          hosts:
            co-infra-1.installation.example.net:
```

```
co_infra:  
  hosts:  
    co-infra-1.installation.example.net:  
co_setup:  
  hosts:  
    co-infra-1.installation.example.net:
```

## ПРИЛОЖЕНИЕ В

### Пример файла inventory Системы редактирования и совместной работы (кластерная установка)

```
all:
  children:

  co:
    children:
      co_etcd:
        hosts:
          etcd-1.installation.example.net:
          etcd-2.installation.example.net:
          etcd-3.installation.example.net:
          etcd-4.installation.example.net:
          etcd-5.installation.example.net:

      co_mq:
        hosts:
          mq-1.installation.example.net:
          mq-2.installation.example.net:
          mq-3.installation.example.net:

      co_cvm:
        hosts:
          cvm-1.installation.example.net:
          cvm-2.installation.example.net:

      co_cu:
        hosts:
          cvm-1.mcs.installation.example.net:
          cvm-2.mcs.installation.example.net:

      co_dcm:
        hosts:
          dcm-1.installation.example.net:
          dcm-2.installation.example.net:
          dcm-3.installation.example.net:

      co_du:
        hosts:
          dcm-1.installation.example.net:
          dcm-2.installation.example.net:
          dcm-3.installation.example.net:

      co_fm:
        hosts:
          fm-1.installation.example.net:
          fm-2.installation.example.net:

      co_jod:
        hosts:
          cvm-1.installation.example.net:
          cvm-2.installation.example.net:

      co_nm:
        hosts:
          nm-1.installation.example.net:
          nm-2.installation.example.net:
```

```
co_pregen:  
  hosts:  
    pregen-1.installation.example.net:  
    pregen-2.installation.example.net:
```

```
co_imc:  
  hosts:  
    imc-1.installation.example.net:  
    imc-2.installation.example.net:  
    imc-3.installation.example.net:
```

```
co_lb_core_auth:  
  hosts:  
    auth-1.installation.example.net:  
    auth-2.installation.example.net:
```

```
co_infra:  
  hosts:  
    log.installation.example.net:
```

```
co_setup:  
  hosts:  
    auth-1.installation.example.net:  
    auth-2.installation.example.net:  
    cvm-1.installation.example.net:  
    cvm-2.installation.example.net:  
    dcm-1.installation.example.net:  
    dcm-2.installation.example.net:  
    dcm-3.installation.example.net:  
    etcd-1.installation.example.net:  
    etcd-2.installation.example.net:  
    etcd-3.installation.example.net:  
    etcd-4.installation.example.net:  
    etcd-5.installation.example.net:  
    fm-1.installation.example.net:  
    fm-2.installation.example.net:  
    imc-1.installation.example.net:  
    imc-2.installation.example.net:  
    imc-3.installation.example.net:  
    log.installation.example.net:  
    mq-1.installation.example.net:  
    mq-2.installation.example.net:  
    mq-3.installation.example.net:  
    nm-1.installation.example.net:  
    nm-2.installation.example.net:  
    pregen-1.installation.example.net:  
    pregen-2.installation.example.net:
```