ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

Система редактирования и совместной работы

Система хранения данных

3.6.1

Архитектура

На 29 листах

Версия документа: 1

Дата публикации: 23.12.2025

Москва
2025

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

Товарные знаки «МойОфис» и «MyOffice» 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.

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

СОДЕРЖАНИЕ

1 Архитектурная схема продукта 5

1.1 Общая схема взаимодействия компонентов 5
1.2 Состав компонентов Системы редактирования и совместной работы 6
1.3 Состав компонентов Системы хранения данных 7

2 Описание архитектуры компонентов 9

2.1 Архитектурная схема Системы редактирования и совместной работы 9
2.2 Описание архитектуры Системы редактирования и совместной работы 10
2.3 Архитектурная схема Системы хранения данных 13
2.4 Описание архитектуры Системы хранения данных 14

3 Типовые схемы установки 16

3.1 Конфигурация без отказоустойчивости 16
3.2 Кластерная отказоустойчивая конфигурация 17
3.3 Требования для кластера с профилем более 5000 пользователей 18
3.4 Расчет требований IOPS 18
3.5 Типовая схема масштабирования 19
3.6 Системные учетные записи 19

4 Дополнительная информация 21

4.1 Описание ролей при расчете аппаратных требований 21
4.2 Пример файла inventory CO (установка standalone) 23
4.3 Пример файла inventory CO (кластерная установка) 25
4.4 Пример файла inventory PGS (установка standalone) 27
4.5 Пример файла inventory PGS (кластерная установка) 28

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ, ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ

В настоящем документе применяют 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, подсистема единого входа (аутентификации и авторизации)
CA	Certificate Authority, удостоверяющий центр для подтверждения подлинности ключей шифрования
CDN	Content Delivery Network, сеть доставки содержимого (в данном случае одна из ролей Auth SSO сервера)
CO	Система редактирования и совместной работы
CU	Converter Unit, сервис конвертирования разных форматов файлов
DNS	Domain Name System, система доменных имен
DU	Document Unit, синоним DCS
ETCD	Распределенная система хранения конфигурации
FCM	Firebase Cloud Messaging, сервис уведомлений мобильных приложений Google, ранее назывался GCM
FQDN	Fully Qualified Domain Name, полностью определенное имя домена
GCM	Google Cloud Messaging, сервис нотификаций мобильных приложений Google, заменен сервисом FCM
Inventory	Файл, содержащий набор управляемых хостов для автоматизации установки и управления конфигурацией для сервиса Ansible
PGS	Система хранения данных
SMTP	Simple Mail Transfer Protocol, протокол передачи почтовых сообщений
SSH	Secure Shell, «безопасная оболочка»
UI	User Interface, пользовательский интерфейс
URL	Uniform Resource Locator, единый указатель ресурса
ОС	Операционная система
ПК	Персональный компьютер
ПО	Программное обеспечение
УЦ	Удостоверяющий центр

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

1 АРХИТЕКТУРНАЯ СХЕМА ПРОДУКТА

1.1 Общая схема взаимодействия компонентов

На рисунке 1 представлена общая схема взаимодействия компонентов продукта.

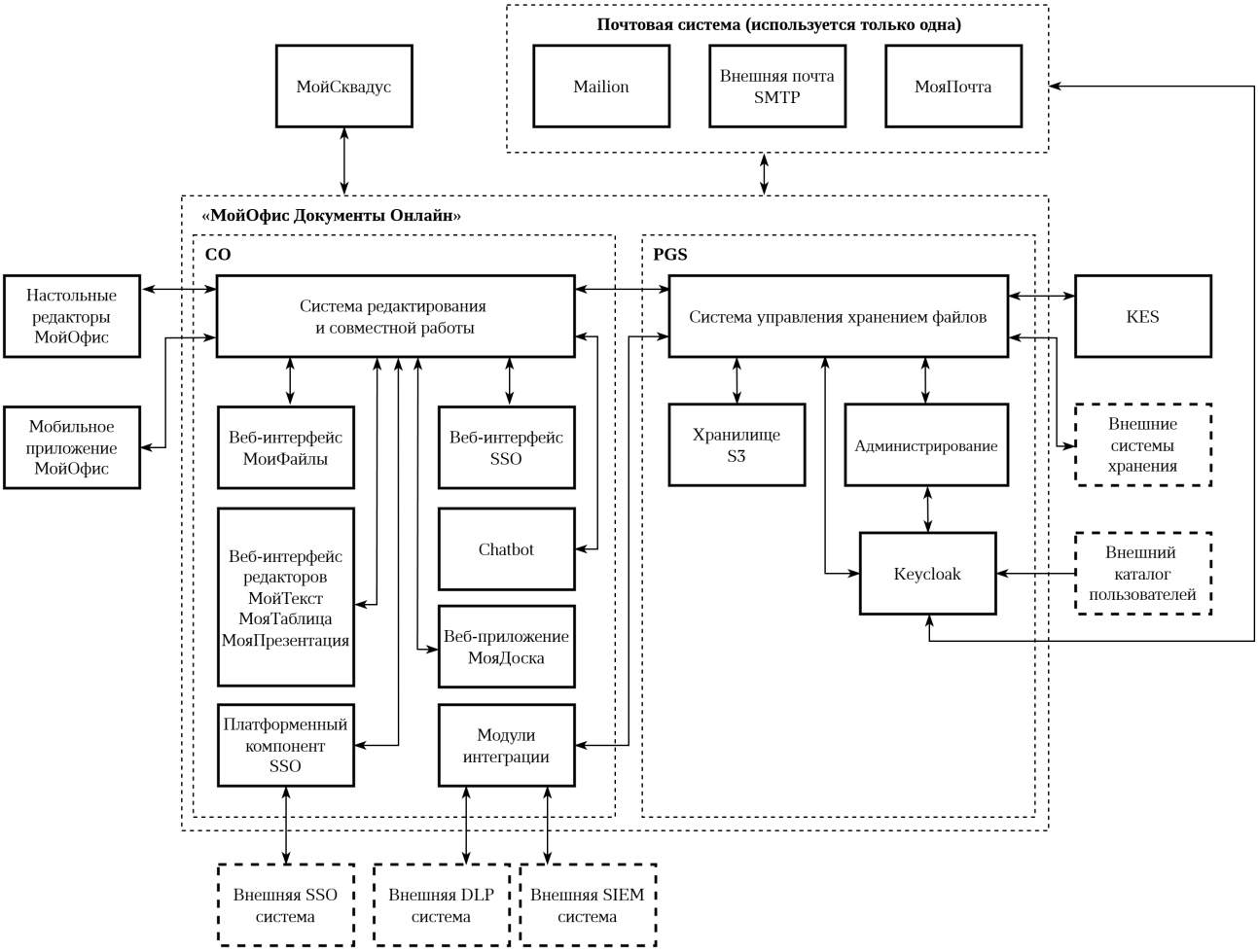

Рисунок 1 — Общая компонентная схема

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

1.2 Состав компонентов Системы редактирования и совместной работы

Состав и назначение компонентов Системы редактирования и совместной работы приведены в таблице 1.

Таблица 1 — Состав и назначение компонентов СО

Наименование компонента	Описание
Система редактирования и совместной работы	Серверный компонент, обеспечивающий: – совместное редактирование документов в Облаке с использованием любой платформы (веб, мобильные или настольные приложения); – взаимодействие клиентских приложений с Системой хранения данных; – интеграцию с почтовыми системами
Веб-интерфейс МоиФайлы	Веб-приложение «МоиФайлы» предназначено для организации структурированного хранения файлов, выполнения операций с файлами и папками, настройки совместного доступа
Веб-интерфейс SSO	Веб-интерфейс предназначен для авторизации, аутентификации в системе и управления профилем пользователя. Главная страница доступа к приложениям (лендинг)
Веб-интерфейс редакторов	Включает в себя следующие редакторы: – веб-приложение «МойТекст» — для создания и форматирования текстовых документов; – веб-приложение «МояТаблица» — для создания электронных таблиц, ведения расчетов, анализа данных и просмотра сводных отчетов; – веб-приложение «МояПрезентация» — для создания, оформления и демонстрации презентаций; – веб-приложение «МояДоска» — для создания виртуальной доски, визуализации идей и проектов на доске с использованием широкого набора инструментов
Мобильные приложения МойОфис	Мобильное приложение «МоиФайлы» предназначено для просмотра и редактирования текстовых документов, электронных таблиц и презентаций, просмотра PDF-файлов, а также доступа к облачным хранилищам на смартфонах и планшетах с ОС Android, iOS и iPadOS. На устройствах под управлением ОС Аврора доступна возможность авторизоваться в продукте и осуществлять просмотр и редактирование файлов, размещенных в хранилище
Платформенный компонент SSO	Сервис, обеспечивающий аутентификацию и авторизацию с помощью внешней SSO-системой

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Наименование компонента	Описание
chatBot	Сервис, обеспечивающий работу виджета Squadus в окне веб-редактора
Модули интеграции	Сервисы, обеспечивающие взаимодействие с внешними системами: DLP, SIEM

Система редактирования и совместной работы предусматривает подключение пользователей настольных редакторов (подробнее см. в документе «Руководство по настройке»).

1.3 Состав компонентов Системы хранения данных

Система управления хранением файлов — серверный компонент, обеспечивающий:

– хранение объектов пользователя;
– хранение общих корпоративных объектов;
– поиск по имени и содержимому;
– выполнение файловых операций для пользователя;
– разграничение прав доступа;
– аутентификацию и валидацию статусов учетных записей пользователей и прав доступа к системе.

В качестве веб-интерфейса используется файловый менеджер.

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Состав и назначение компонентов Системы хранения данных приведены в таблице 2.

Таблица 2 — Состав и назначение компонентов Системы хранения данных

Наименование компонента	Описание
Keycloak	Сервис содержит список пользователей системы и их атрибуты. Предусмотрена возможность синхронизации пользователей из внешних каталогов (Active Directory). Keycloak обеспечивает хранение списка тенантов и их атрибутов продукта. Для управления параметрами установки от имени суперадминистратора используется графический интерфейс
Хранилище	Тип файлового хранилища определяется на этапе первичной установки компонента. Используется объектное хранилище типа S3 (minIO)*.
Администрирование	Серверный компонент, обеспечивающий: – управление политиками безопасности и другими настройками тенанта (организации); – ролевую модель доступа к системе; – управление доменами, пользователями, группами пользователей, общими папками, публичными ссылками от имени администратора системы; – настройку доступности функциональных блоков продукта для пользователей. Для управления компонентом используется веб-интерфейс
* — *Для хранилища типа S3 предусмотрена интеграция с другим S3-хранилищем в уже существующей инфраструктуре.*	

Для обеспечения уровня безопасности в Системе хранения данных предусмотрена интеграция с Kaspersky Endpoint Security (KES) (подробнее см. в документе «Руководство по настройке»).

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

2 ОПИСАНИЕ АРХИТЕКТУРЫ КОМПОНЕНТОВ

2.1 Архитектурная схема Системы редактирования и совместной работы

На рисунке 2 представлена схема взаимодействия подсистем Системы редактирования и совместной работы. Прямоугольниками выделены кластерные решения для сервисов `etcd`, `redis`, `rabbitmq`.

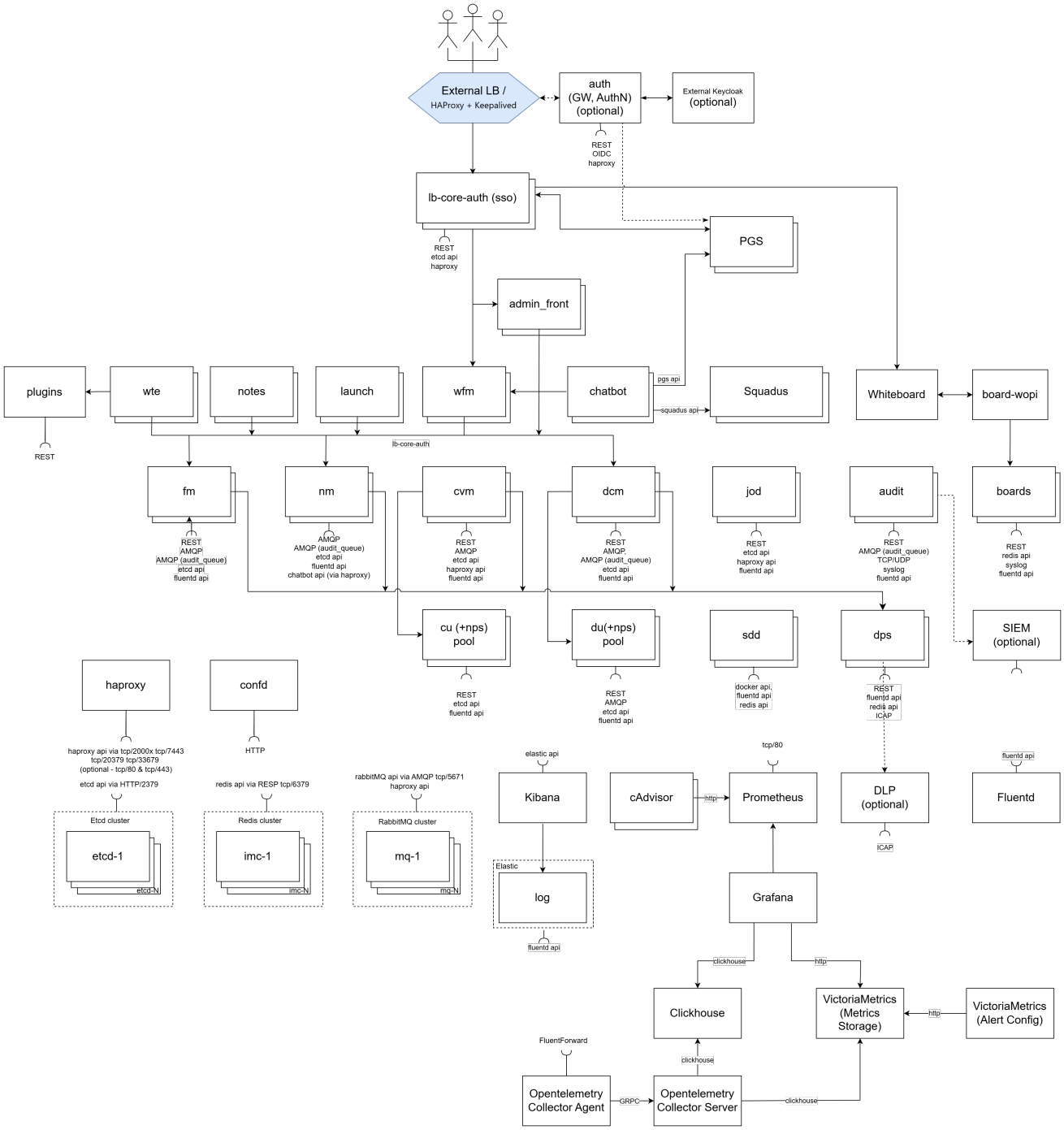

Рисунок 2 — Архитектурная схема Системы редактирования и совместной работы

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

2.2 Описание архитектуры Системы редактирования и совместной работы

Описание подсистем и сервисов Системы редактирования и совместной работы приведено в таблице 3.

Таблица 3 — Описание подсистем и сервисов Системы редактирования и совместной работы

Наименование подсистемы или сервиса	Описание
Auth	Компонент, состоящий из 2 сервисов: – GW — API-шлюз; – Authn — сервис аутентификации. Сервисы работают в тандеме и обеспечивают возможность интеграции с External Keycloak (SSO-системой)
Lb-core-auth (nginx-wfe)	Расширенный веб-сервер NGINX с поддержкой Lua. Отвечает за авторизацию, доступность API, балансировку
HAProxy	Балансировщик нагрузки между внутренними сервисами
FM	File Manager, сервис файлового менеджера. Отвечает за создание, удаление, добавление файлов, предоставление доступа другим пользователям
NM	Notification Manager, сервис управления уведомлениями
CVM	Conversion Manager, сервис управления конвертированием файлов. При конвертации документов использует пул CU юнитов или Jod
CU	Conversion Unit, экземпляр процесса конвертора различных форматов файлов (офисных или изображений)
DCM	Document Collaboration Manager, сервис управления редактированием документов. При редактировании документов использует пул DU юнитов
DPS	Data Protection Service, сервис сбора и передачи событий по REST API во внешнюю DLP-систему по протоколу ICAP
DLP	Data Loss Prevention, внешняя система защиты от потери данных
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

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Наименование подсистемы или сервиса	Описание
Audit	Сервис аудита, получает события аудита из сервисов и Системы хранения данных через RabbitMQ. Выполняет отправку событий аудита (TCP/UDP) в syslog или SIEM (не входит в стандартную поставку)
Elastic	Elastic (Opendistro Elastic Search), поисковый и аналитический сервис. Отвечает за сбор, обработку и хранение логов всех сервисов
Fluentd	Сервис, который собирает логи от всех сервисов и передает их в Elastic Search
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	noSQL in-memory 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
NPS	Native Process Service, сервис управления CU, DU
SDD	Service Detector Docker, обнаружение сервисов. Опрашивает сервисы NPS через Docker API и сокет
Prometheus	Инструмент сбора метрик со всех сервисов
Grafana	Инструмент визуализации метрик приложения. Получает метрики из Prometheus
WFM	Web file manager (SPA веб-приложение). Веб-приложение файлового менеджера (виджет вложений). Отвечает за операции с файлами (кроме редактирования) и просмотр файлов по публичным ссылкам
Chatbot	NodeJs сервис. Отвечает за создание чата по документу, добавления пользователей и обновления списка пользователей в чате
WTE	Web text editors (веб-приложение редактора). Отвечает за редактирование и чтение файлов
External Keycloak	Внешняя SSO-система, сохраняющая в себе настройки системы, данные по тенантам и пользователям
Whiteboard	Веб-клиент досок. Отвечает за интеграцию приложения доски с продуктом

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Наименование подсистемы или сервиса	Описание
board-wopi	Приложение досок. Отвечает за редактирование содержимого доски
boards	Сервис управления редактирования досок
plugins	Сервис для работы с плагинами
cAdvisor	Сервис мониторинга контейнеров, собирает метрики ресурсов (CPU, RAM, сеть, диски) и статистику по работающим контейнерам. Используется в связке с Prometheus, Grafana
confd	Сервис автоматической генерирации конфигурации из шаблонов (например, для Nginx, HAProxy, etcd). Интегрируется с хранилищами ключ-значение(etcd, env-переменные).
launch	Веб-клиент приложений МоиДоски, МоиТаблицы, МоиПрезентации, МойТекст
notes	Веб-клиент приложения МоиЗаметки
admin_front	Веб-клиент приложения Администрирование
SIEM	SIEM (Security information and event management) — система регистрации событий аудита, которые получает от сервиса audit. Используется опционально.
OpenTelemetry Collector as an Agent	OpenTelemetry Collector (в режиме агента) при котором сервис устанавливается на сервере (или в sidecar-контейнер). Служит для сбора, обработки и экспорта телеметрических данных (трассировок, метрик, логов) с локальных приложений или сервисов
OpenTelemetry Collector as a Server	OpenTelemetry Collector (в режиме централизованного сервиса) для приема, обработки и маршрутизации телеметрических данных (трассировок, метрик, логов) от разных источников (агентов, приложений или других коллекторов). Выполняет функцию промежуточного узла перед отправкой данных в бэкенд-системы
ClickHouse	Высокопроизводительная колоночная СУБД с открытым исходным кодом, разработанная для аналитики (OLAP) и работы с большими данными в реальном времени
VictoriaMetrics	Высокопроизводительная, экономичная и масштабируемая база данных для хранения и анализа метрик временных рядов (time-series data)
HAProxy + Keepalived	Сервисы Keepalived и HAProxy служат для балансировки между серверами с ролью `co_lb_core_auth`

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

2.3 Архитектурная схема Системы хранения данных

На рисунке 3 представлена схема взаимодействия компонентов Системы хранения данных.

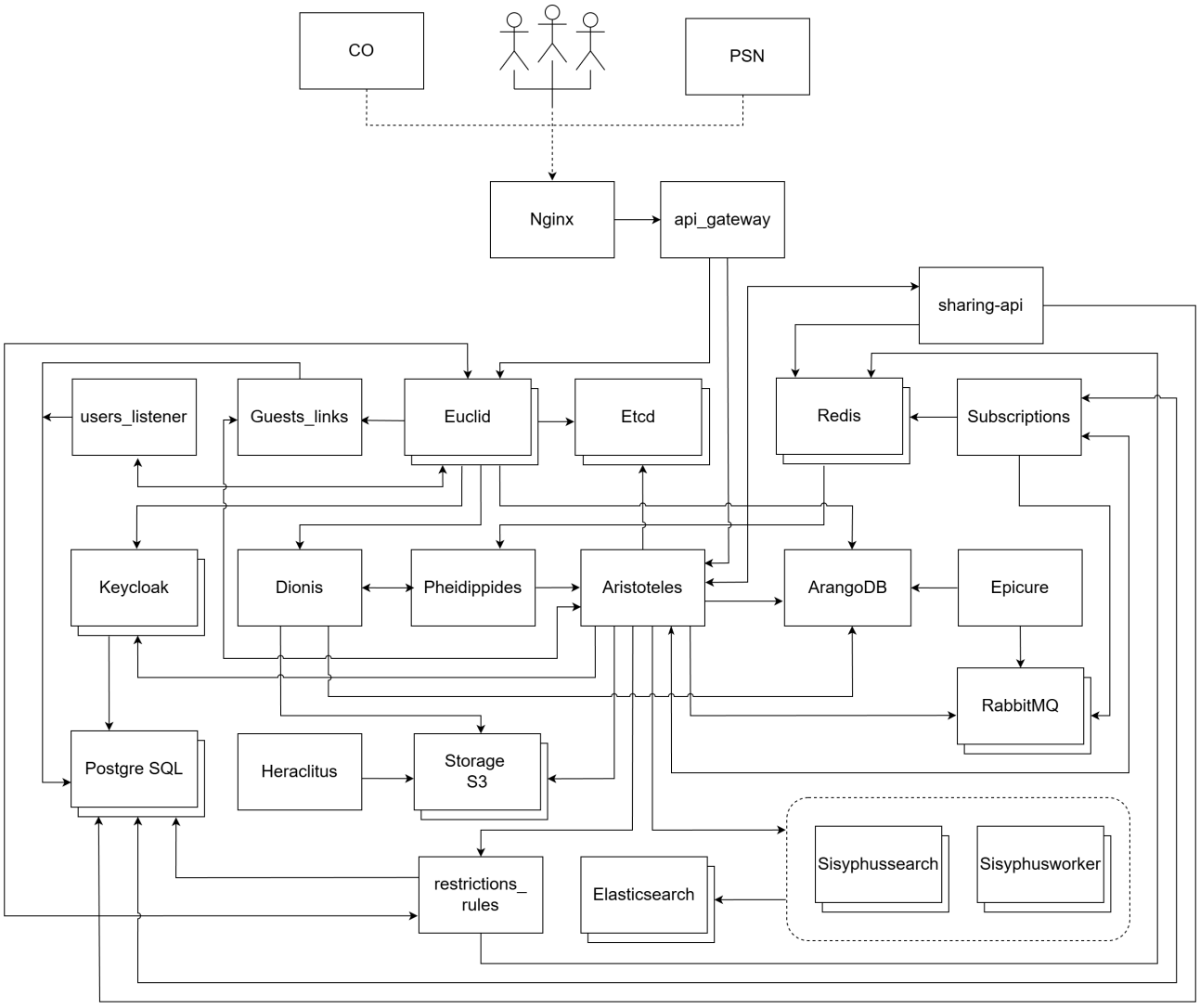

Рисунок 3 — Архитектурная схема Системы хранения данных

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

2.4 Описание архитектуры Системы хранения данных

Описание подсистем и сервисов Системы хранения данных приведено в таблице 4.

Таблица 4 — Описание подсистем и сервисов Системы хранения данных

Наименование сервиса	Описание
Arangodb	База данных, содержащая метаданные файлов (в т. ч. информацию о владельце документа, правах доступа и пр.)
Aristoteles	Сервер приложений, выступающий backend-частью для компонентов Системы редактирования и совместной работы в части выполнения файловых операций, разграничения прав доступа, версионирования, фиксации истории событий по объектам
Dionis	Сервис, отвечающий за удаление и переназначение прав доступа для объектов пользователей
Elasticsearch	Сервис, отвечающий за поиск по содержимому по хранящимся файлам
Epicure	Сервис формирования и отправки сообщений безопасности с последующей отправкой в аудит-системы (SIEM)
Etcd	Сервис, содержащий конфигурацию приложений, при кластерном развертывании также используется сервисом Postgres для создания кластера
Euclid	Rest API сервис, выступающий backend-частью для компонента polemon (веб-администрирование). Отвечает за администрирование пользователей в системе
Guests_links	Сервис, осуществляющий создание, обновление, удаление гостевых ссылок
Heraclitus	Сервис очистки архивных данных, удаленных пользователями из корзины. Может настраивать сроки хранения архивных данных и автоудаления их с диска по заданному расписанию
Keycloak	Сервис SSO, сохраняет в себе настройки системы, данные по тенантам и пользователям
Nginx	Прокси-сервис, обеспечивающий доступ до: rabbitmq, aristotels, euclid
Pheidippides	Сервис, осуществляющий обработку событий в Redis каналах (автоматическая блокировка IP-адресов/публичных ссылок)
Postrgres	PostgreSQL, база данных для сервиса авторизации Keycloak

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Наименование сервиса	Описание
RabbitMQ	Очередь сообщений. Используется для передачи документов в elasticsearch для поиска по содержимому документов и для передачи межкомпонентных уведомлений от Системы хранения данных в Систему редактирования и совместной работы об изменении настроек хранилища
Redis	База данных «ключ-значение» для неперсистентных данных (в основном используется для хранения токенов и других авторизационных данных)
sharing-api	Сервис, осуществляющий создание и управление прямыми ссылками, проверку доступа к файлам по прямым ссылкам
Sisyphus_ sisyphussearch	Сервис, осуществляющий поиск по содержимому документов в elasticsearch
Sisyphus_ sisyphusworker	Сервис, осуществляющий передачу файлов из rabbitMQ в elasticsearch
Storage S3	Блок storage осуществляет хранение файлов системы. В качестве хранилища S3 в проекте используется minIO.
subscriptions	Сервис, отвечающий за функционал подписок и уведомлений
api_gateway	Внутренний прокси-сервер PGS. Расположен после Nginx и перенаправляет запросы с постфиксом /pgsapi, /adminapi в соответствующие сервисы
user_listener	Сервис для прослушивания событий в PostgreSQL. Используется при интеграции с внешними каталогами (AD, FreeIPA, OpenLDAP). Отслеживает события изменений статуса пользователя и отправляет запросы в Euclid по http для обновления статуса пользователя
restrictions_rules	Сервис с правилами запретов и исключений. Интерфейс управления доступен из административной панели

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

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.

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

– всего пользователей — 999;

– количество одновременно активных пользователей — 500;

– количество документов, редактируемых одновременно — 45.

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

Пример расчета аппаратных требований для установки на двух виртуальных серверах приведен в таблице 5.

Таблица 5 — Аппаратные требования для конфигурации без отказоустойчивости

Роли серверов	CPU, vCPU	RAM, Гбайт	SSD, Гбайт
operator*	1	4	50
Все роли Системы хранения данных	8	20	100 **
Все роли Системы редактирования и совместной работы	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.*
*** — зависит от дисковой квоты*

Примеры файла inventory для Системы редактирования и совместной работы представлены и Системы хранения данных представлены в разделе Дополнительная информация.

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

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 рекомендуемые требования для развертывания системы в режиме кластера со следующим профилем эксплуатации:

– всего пользователей — 5000;

– количество одновременно активных пользователей — 2500;

– количество документов, редактируемых одновременно — 315.

Пример расчета аппаратных требований для отказоустойчивой установки приведен в таблице 6.

Таблица 6 — Аппаратные требования для кластерной отказоустойчивой конфигурации

Наименование роли*	**Количество серверов**	**CPU, vCPU**	**RAM, Гбайт**	**SSD, Гбайт**	**HDD, Гбайт**
operator	1	4	8	0	50
LB	2	2	4	50	0
core+etcd	2	10	24	100	0
infra	1	4	16	100	0
mq+imc+etcd	3	6	12	50	0
PGS	3	6	16	0	70
PGS-STORAGE **	4	8	16	0	11155
PGS-A ***	1	6	12	0	3 350
* — *описания ролей представлены в разделе Дополнительная информация* ** — *не менее 2-х серверов* *** — *роль используется, если не подключено внешнее S3 хранилище (MinIO)*					

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

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 (для этого роль `etcd` размещается на сервере роли `core`);
– роли `cvm`/`cu-pool` и `dcm`/`du-pool` в такой конфигурации объединены внутри роли core+etcd;
– роль LB — внешний балансировщик.

Примеры файла inventory для Системы редактирования и совместной работы представлены и Системы хранения данных представлены в разделе Дополнительная информация.

3.3 Требования для кластера с профилем более 5000 пользователей

Для кластерной установки, на которой планируется работа более 5000 пользователей, необходимо обратиться к вендору для расчета размеров серверных ресурсов и получения рекомендаций по объединению ролей.

3.4 Расчет требований IOPS

Требования IOPS к различным ролям представлены в таблице 7.

Таблица 7 — Требования IOPS для ролей

Наименование роли *	**Среднее значение IOPS**	**Максимальное значение IOPS****
PGS-APP	15	300
PGS-A	30	100
PGS-STORAGE ***	30	100
*— описание ролей представлены в разделе Дополнительная информация ** — значения полученные при тестах эквивалентных работе системы при развертывании PGS на 80к пользователей ***— данный расчет приведен при использовании MinIO с erasure set = 16.		

Для работы сервиса пропускная способность дисковой подсистемы должна быть не ниже среднего значения IOPS, для комфортной работы пропускная способность должна быть выше максимального требования IOPS.

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

3.5 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. В Системе хранения данных переход обеспечивается с помощью предварительного резервирования баз данных.

Для Системы редактирования и совместной работы в первую очередь следует масштабировать узлы кластера с ролями:

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

Для Системы хранения данных в первую очередь следует масштабировать серверы с ролью ArangoDB (компонента PGS). Горизонтальное масштабирование невозможно, при увеличении нагрузки следует увеличить аппаратные мощности серверов.

3.6 Системные учетные записи

В таблице 8 и 9 представлены системные учетные записи, необходимые для связи сервисов системы.

Таблица 8 — Системные учетные записи CO

Имя учетной записи	Имя сервиса	Описание
app-co	PGS	Получение настроек тенантов
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

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Имя учетной записи	Имя сервиса	Описание
root	RabbitMQ	Пользователь RabbitMQ для управления очередями и конфигурации во время деплоя
rabbitmq	RabbitMQ	Пользователь в PGS RabbitMQ для федерации с CO RabbitMQ

Таблица 9 — Системные учетные записи PGS

Имя учетной записи	Имя сервиса	Описание
app-co	Keycloak	Получение настроек тенантов
pgs	Keycloak	Управление тенантами/ Учетная запись в keycloak
root	Arango	Пользователь ARANGO DB
keycloak	Postgres	Пользователь СУБД Postgres
admin	Grafana	Пользователь Grafana UI
rabbitmq	RabbitMQ	Пользователь в RabbitMQ

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

4 ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

4.1 Описание ролей при расчете аппаратных требований

Таблица А.1 — Роли для установки standalone

Наименование	Описание
PGS	Содержит службы хранилища, БД пользователей, администрирования, API хранилища
CO	Содержит службы SSO, веб-редакторов, collaboration API

Таблица А.2 — Роли для кластерной установки Системы хранения данных

Наименование	Описание
PGS	Сервер вычислений, обработки запросов, API, keycloak, внутренний балансировщик
PGS-STORAGE	Сервер хранения данных
PGS-A	Сервер индексного поиска, хранения кэш, конфигурации, очередей (elasticsearch, etcd, arangodb_agent, redis, rabbitmq)

Таблица А.3 — Роли для кластерной установки Системы редактирования и совместной работы

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
etcd	Подсистема конфигурации
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
core-fm	Подсистема сервиса файлового API
core-nm	Подсистема сервиса push-уведомлений

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

Наименование	Описание
imc	Сервер кеширования 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.4 — 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)

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

4.2 Пример файла inventory CO (установка standalone)

```
all:
  children:

    co:
      children:
        co_audit:
          hosts:
            co-infra-1.installation.example.net:
        co_authn:
          hosts:
            co-infra-1.installation.example.net:
        co_boards:
          hosts:
            co-infra-1.installation.example.net:
        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_dps:
          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_gateway:
          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_imc:
          hosts:
            co-infra-1.installation.example.net:
        co_lb_core_auth:
          hosts:
            co-infra-1.installation.example.net:
        co_plugins:
          hosts:
            co-infra-1.installation.example.net:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

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

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

4.3 Пример файла inventory CO (кластерная установка)

```
all:
  children:

    co:
      children:
        co_audit:
          hosts:
            co-audit-[1:2].installation.example.net:
        co_authn:
          hosts:
            co-authn-[1:2].installation.example.net:
        co_boards:
          hosts:
            co-auth-[1:2].installation.example.net:
        co_chatbot:
          hosts:
            co-chatbot-[1:2].installation.example.net:
        co_etcd:
          hosts:
            co-etcd-[1:3].installation.example.net:
        co_mq:
          hosts:
            co-imc-mq-[1:3].installation.example.net:
        co_cvm:
          hosts:
            co-cvm-[1:2].installation.example.net:
        co_cu:
          hosts:
            co-cvm-[1:2].installation.example.net:
        co_dcm:
          hosts:
            co-dcm-[1:2].installation.example.net:
        co_dps:
          hosts:
            co-dps-[1:2].installation.example.net:
        co_du:
          hosts:
            co-dcm-[1:2].installation.example.net:
        co_fm:
          hosts:
            co-fm-[1:2].installation.example.net:
        co_gateway:
          hosts:
            co-gateway-[1:2].installation.example.net:
        co_jod:
          hosts:
            co-jod-[1:2].installation.example.net:
        co_nm:
          hosts:
            co-nm-[1:2].installation.example.net:
        co_imc:
          hosts:
            co-imc-mq-[1:3].installation.example.net:
        co_lb_core_auth:
          hosts:
            co-auth-[1:2].installation.example.net:
        co_plugins:
          hosts:
            co-auth-1.installation.example.net:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

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

    co_setup:
      hosts:
        co-auth-[1:2].installation.example.net:
        co-chatbot-1.installation.example.net:
        co-cvm-[1:2].installation.example.net:
        co-dcm-[1:2].installation.example.net:
        co-etcd-[1:3].installation.example.net:
        co-fm-[1:2].installation.example.net:
        co-imc-mq-[1:3].installation.example.net:
        co-infra-1.installation.example.net:
        co-jod-[1:2].installation.example.net:
        co-nm-[1:2].installation.example.net:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

4.4 Пример файла inventory PGS (установка standalone)

```
all:
  children:
    pgs:
      children:
        pythagoras:
          hosts:
            host.example.com:
        keycloak:
          hosts:
            host.example.com:
        arangodb:
          hosts:
            host.example.com:
              volume_device_arangodb: "False"
              volume_device_arangodb_path: "/dev/disk/by-uuid/<UUID>"
        arangodb_agent:
          hosts:
            host.example.com:
        search:
          hosts:
            host.example.com:
              volume_device_elasticsearch: "False"
              volume_device_elasticsearch_path: "/dev/disk/by-uuid/<UUID>"
        redis:
          hosts:
            host.example.com:
        rabbitmq:
          hosts:
            host.example.com:
        postgres:
          hosts:
            host.example.com:
              volume_device_postgres: "False"
              volume_device_postgres_path: "/dev/disk/by-uuid/<UUID>"
        etcd:
          hosts:
            host.example.com:
        nginx:
          hosts:
            host.example.com:
        infrastructure:
          hosts:
            host.example.com:
    storage:
      hosts:
        host.example.com:

    pgs_setup:
      hosts:
       host.example.com:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

4.5 Пример файла inventory PGS (кластерная установка)

```
all:
  children:
    pgs:
      children:
        pythagoras:
          hosts:
            host1.example.com:
            host2.example.com:
        keycloak:
          hosts:
            host1.example.com:
            host2.example.com:
        arangodb:
          hosts:
            host1.example.com:
              volume_device_arangodb: False
              volume_device_arangodb_path: "/dev/disk/by-uuid/<UUID>"
            host2.example.com:
              volume_device_arangodb: False
              volume_device_arangodb_path: "/dev/disk/by-uuid/<UUID>"
        arangodb_agent:
          hosts:
            host1.example.com:
              volume_device_agent: False
              volume_device_agent_path: "/dev/disk/by-uuid/<UUID>"
            host2.example.com:
              volume_device_agent: False
              volume_device_agent_path: "/dev/disk/by-uuid/<UUID>"
            host3.example.com:
              volume_device_agent: False
              volume_device_agent_path: "/dev/disk/by-uuid/<UUID>"
        search:
          hosts:
            host1.example.com:
              volume_device_elasticsearch: False
              volume_device_elasticsearch_path: "/dev/disk/by-uuid/<UUID>"
            host2.example.com:
              volume_device_elasticsearch: False
              volume_device_elasticsearch_path: "/dev/disk/by-uuid/<UUID>"
            host3.example.com:
              volume_device_elasticsearch: False
              volume_device_elasticsearch_path: "/dev/disk/by-uuid/<UUID>"
        redis:
          hosts:
            host1.example.com:
            host2.example.com:
            host3.example.com:
        rabbitmq:
          hosts:
            host1.example.com:
            host2.example.com:
            host3.example.com:
        etcd:
          hosts:
            host1.example.com:
            host2.example.com:
            host3.example.com:
        nginx:
          hosts:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025

```
            host1.example.com:
            host2.example.com:
        postgres:
          hosts:
            host1.example.com:
              volume_device_postgres: False
              volume_device_postgres_path: "/dev/disk/by-uuid/<UUID>"
            host2.example.com:
              volume_device_postgres: False
              volume_device_postgres_path: "/dev/disk/by-uuid/<UUID>"
        infrastructure:
          hosts:
            host3.example.com:
        co_lb:
          hosts:
            co-lb1.example.com:
            co-lb2.example.com:
        co_auth:
          hosts:
            co-auth1.example.com:
            co-auth2.example.com:
    storage: #3 minimum
      hosts:
        host1.example.com:
        host2.example.com:
        host3.example.com:

    pgs_setup:
      hosts:
        host1.example.com:
        host2.example.com:
        host3.example.com:
```

© ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ», 2013–2025