

Архитектура VK Почта

Архитектура VK Почта

Оглавление

Введение	3
Сокращения и определения	3
Архитектура Системы	3
Возможные интеграции	4
Верхнеуровневая архитектура Системы	5
Компонетная архитектура Системы	6
Сервисы системы	7
Сервисы pub	7
Сервис fedman	8
Сервисы аторизации и аутентификации	8
Сервисы МТА	9
Сервис mailapi	10
Сервис imapd	10
Сервисы транспортных правил	10
Сервисы резервного копирования	11
Сервисы миграции	11
Сервис bookster	19
Сервисы поиска	19
Сервис streamer	22
Сервис deliveryd	22
Сервисы аудита	22
Сервисы мониторинга	23
Сервисы хранения данных	24
Технические требования	24

Введение

В настоящем документе приведено описание архитектуры корпоративной электронной почты Почта VK WorkSpace.

Данный документ предназначен для технических специалистов, которые внедряют или поддерживают приведенные технологические решения:

- системные администраторы;
- специалисты по информационной безопасности.

Документ предполагает наличие технических знаний в области проектирования систем, однако будет полезен и специалистам бизнес-подразделений.

Сокращения и определения

Наименование	Определение
СУБД	Система управления базами данных
ЦОД	Центр обработки данных
Сниппет	Фрагмент исходного текста, применяемый в поисковых запросах
AD	Active Directory - служба каталогов пользователей
LDAP	Lightweight Directory Access Protocol - облегченный протокол доступа к каталогам, открытый стандартизированный протокол, применяемый для работы с различными реализациям служб каталогов, в том числе и Active Directory

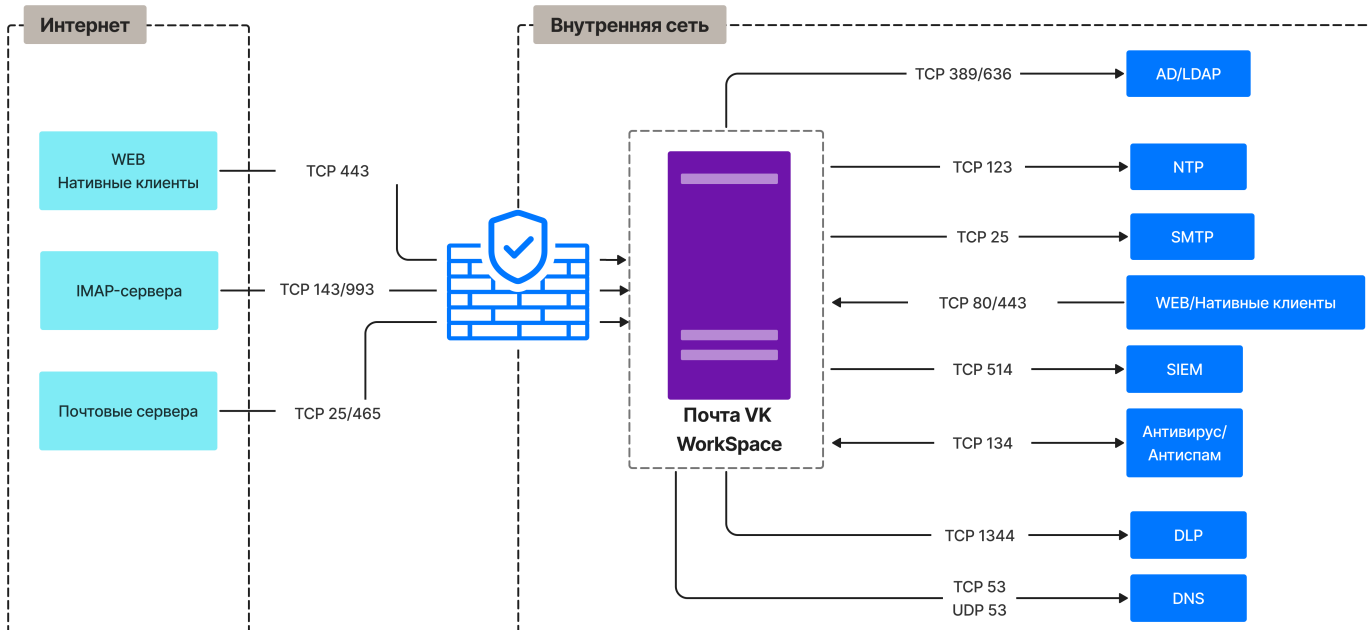
Архитектура Системы

Почта VK WorkSpace может быть установлена как на площадке заказчика, так и в облачном сервисе VK Cloud Solutions.

Архитектура системы позволяет выполнить установку как в полностью закрытый контур для работы только внутри корпоративной сети, так и в контур с возможностью работы из внешних сетей.

Возможные интеграции

На схеме ниже представлены возможные интеграции с Почта VK WorkSpace:



LDAP, AD

Система работает со службами каталогов по LDAP, клиент предоставляет только путь и пароль на чтение. Из инфраструктуры клиента в систему выгружаются и сохраняются все пользователи, группы и группы рассылок, организационно-штатная структура. Администратор настраивает синхронизацию, и дальнейшая аутентификация происходит в AD через LDAP — в почтовой системе пароли не хранятся.

В дополнение к данным из AD в почтовой системе можно вручную создавать пользователей, импортировать данные из CSV-файла.

SIEM

Система управления информационной безопасностью и событиями безопасности. Сервисы Почта VK WorkSpace логируют события безопасности, которые могут быть интегрированы с корпоративной подсистемой SIEM. Для интеграции с SIEM используется сервис управления логами Rsyslog, передача логов поддерживается по протоколам TCP, UDP.

DLP

Интеграция с DLP-системами возможна в разрыв с MX. В почтовой системе MX настраивается на DLP-систему (собственные базовые проверки отключаются), в результате осуществляется пересылка всех писем из DLP-системы в почтовую систему (даже тех писем, которые отправлены внутри почтовой системы).

Антивирус/Антиспам

Используются внешние решения (например, DrWeb, Kaspersky) с целью экономии ресурсов клиента. (Встроенная антивирус / антиспам-система была бы очень ресурсоемка для клиентов, занимает много места, требует большого трафика входящей почты для качественного анализа, что не всегда возможно у клиента.)

NTP

Используется для синхронизации времени. Возможно использование внешних серверов, если нет сложностей с прохождением сетевых фильтров.

DNS

Используется для соотношения IP-адресов и доменных имен, находящихся в базе данных сервера.

IMAP-сервера

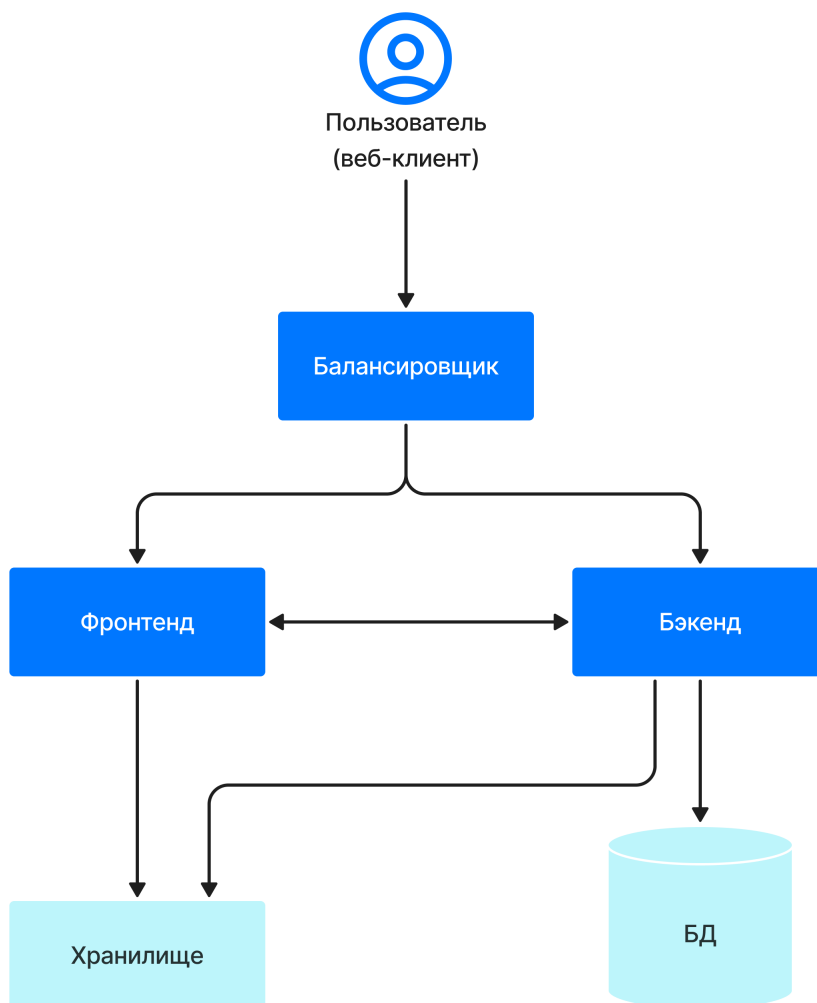
Используется для взаимодействия с внешними почтовыми серверами для отправки/получения и синхронизации почты.

Почтовые сервера/SMTP

Используется для взаимодействия с внешними или внутренними корпоративными почтовыми серверами для отправки/получения почты.

Верхнеуровневая архитектура Системы

Верхнеуровневая архитектура системы представлена на рисунке ниже:



Запросы пользователей поступают на балансировщик, который отвечает за прием и распределение пользовательских запросов между фронтендом и бэкендом, в свою очередь балансировщик отправляет

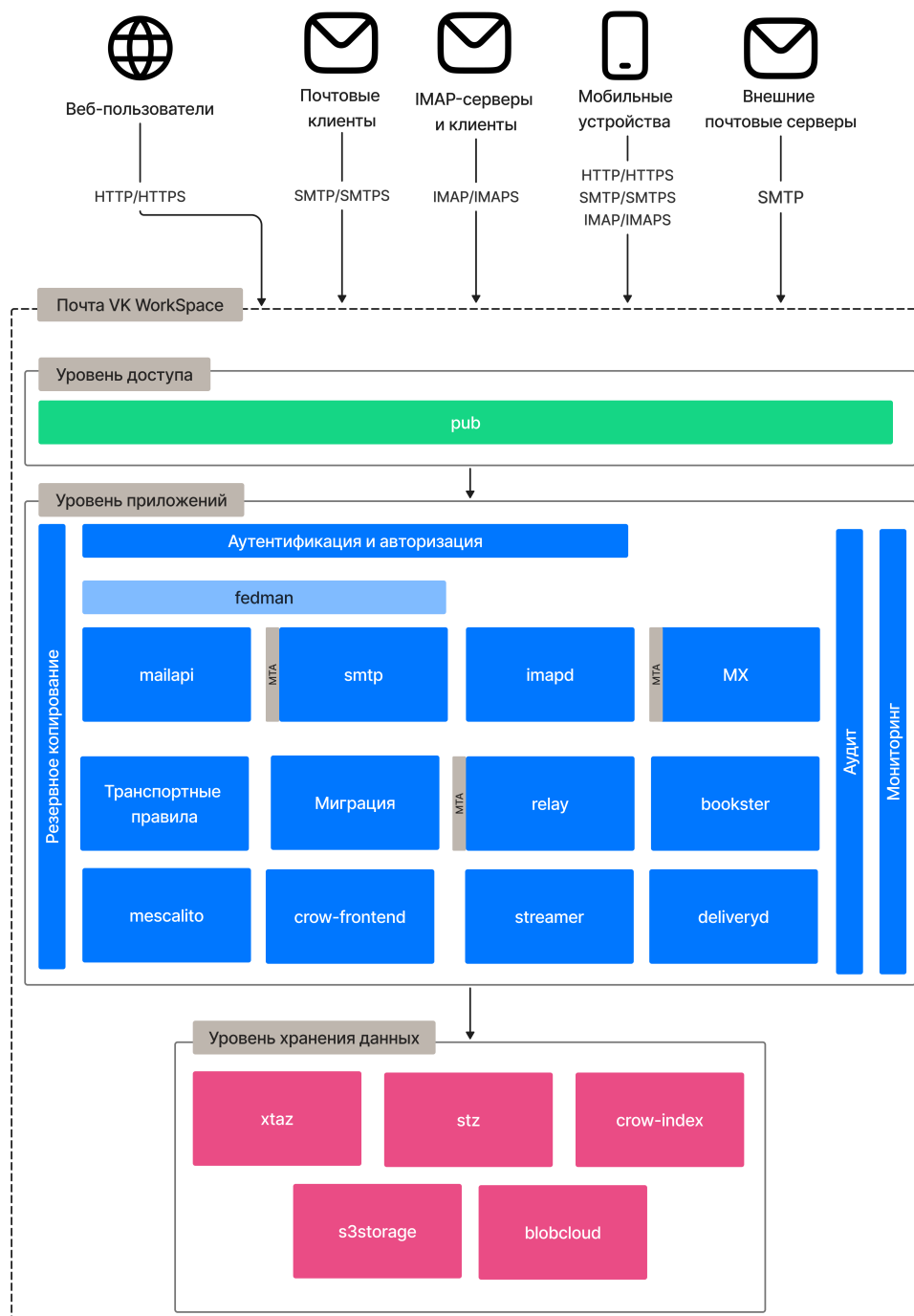
запросы на отдачу динамически генерируемых веб-страниц и статических файлов для передачи в веб-клиент.

Фронтенд и Бэкенд продукта Почта VK WorkSpace состоит из множества различных компонентов (будут рассмотрены ниже), которые отвечают за определенную часть бизнес-логики.

Архитектура продукта Почта VK WorkSpace позволяет выполнить масштабирование — в зависимости от потребностей клиента. При необходимости масштабирования хранилищ и баз данных в Почта VK WorkSpace предусмотрена репликация и шардирование баз данных.

Компонетная архитектура Системы

Схема компонентов продукта Почта VK WorkSpace представлена на рисунке ниже:



На схеме компоненты Почта VK WorkSpace разделены на три логических уровня:

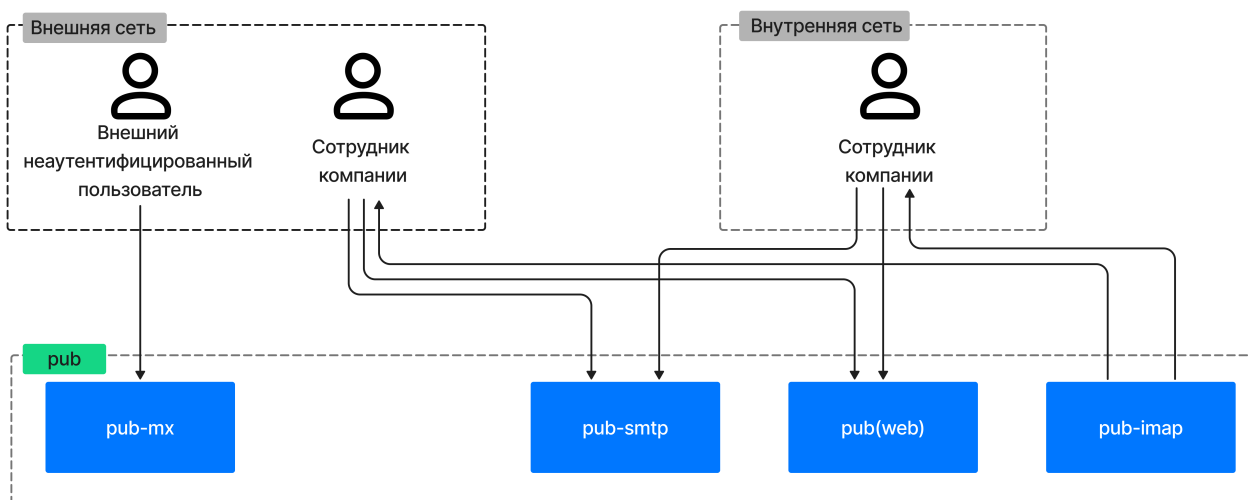
- Уровень клиентского доступа — отвечает за прием всех форм клиентских подключений (запросов), а также выполняет роль бансировки нагрузки и маршрутизации запросов. Подключение клиентов напрямую к уровню приложений и хранению данных — исключено.
- Уровень приложений — отвечает за реализацию бизнес-логики, обеспечивает контроль прав доступа.
- Уровень хранения данных и файлов — отвечает за организацию надежного хранения информационных ресурсов и предоставления гарантированного доступа к ним. Подключение к данному уровню обеспечивается только с уровня приложений.

Сервисы системы

Сервисы pub

Сервисы компонента **pub** являются входной точкой в Почту VK WorkSpace и выполняют различные проверки входных и выходных данных, в том числе авторизуют запросы пользователей.

Общая схема сервисов компонента **pub** приведена на рисунке ниже:



Подробное описание потоков данных приведено в разделе **Диаграммы потоков данных**.

Описание сервисов компонента **pub** приведено в таблице ниже:

Наименование сервиса	Назначение
pub (web)	Прокси-сервис для почтового трафика при отправке почты через веб-клиента, для внешних и внутренних пользователей

Наименование сервиса	Назначение
pub-mx	Прием и фильтр SMTP-запросов от внешних неаутентифицированных пользователей
pub-imap	Прокси почтового трафика по IMAP-протоколу для отправки/получения и синхронизации почты
pub-smtp	Прием и фильтр SMTP-запросов от внутренних и внешних аутентифицированных пользователей

Сервис fedman

⚠ Важно

Сервис **fedman** используется опционально и находится в статусе **deprecated**.

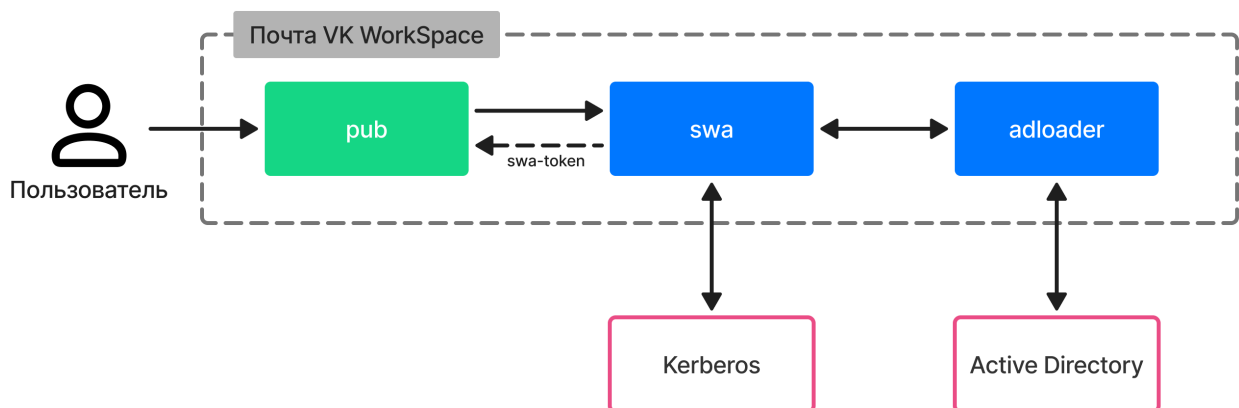
Применение fedman — работа системы Почта VK WorkSpace при размещении в двух и более ЦОД, в рамках одного Заказчика. При этом в каждом ЦОД выполняется самостоятельная отказоустойчивая инсталляция продукта Почта VK WorkSpace и которая обслуживает пользователей, относящихся к каждой из инсталляций. В каждом ЦОД создаются самостоятельные кластеры.

Сервис fedman применяется только для почтового трафика.

В качестве СУБД сервис fedman использует Tarantool. Создается кластер Tarantool в режиме синхронной репликации.

Сервисы аторизации и аутентификации

Общая схема сервисов приведена на рисунке ниже:



Описание сервисов авторизации и аутентификации приведено в таблице ниже:

Наименование сервиса	Назначение
swa	Группа сервисов аутентификации (clauth, GoCube, UMI) выдача, хранение и валидация сессионных токенов для пользователей
adloader	Сервис интеграции с LDAP – синхронизирует списки пользователей и групп рассылок, проверяет пароль в LDAP

Последовательность прохождения аутентификации и авторизации пользователя:

Примечание

Под пользователями в данном случае подразумеваются IMAP, SMTP и HTTP-запросы.

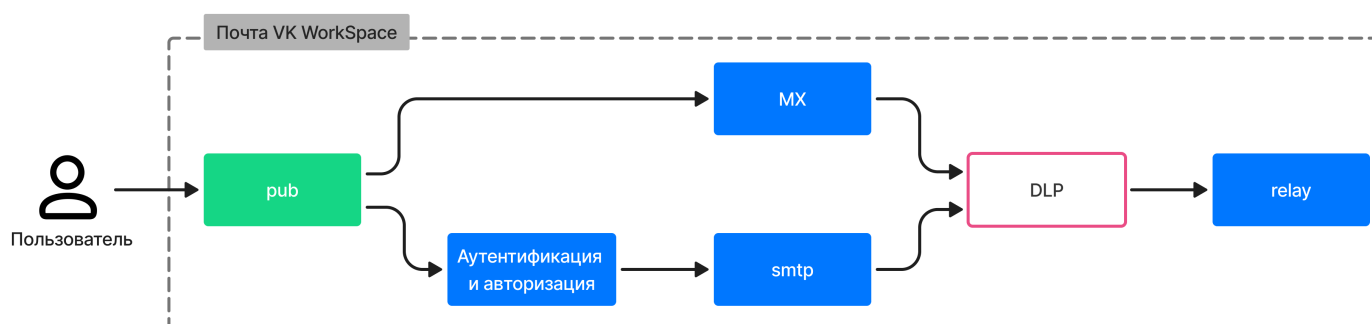
1. Пользователь отправляет запрос на аутентификацию по протоколу IMAP/SMTP/HTTP.
2. **Pub** идентифицирует запрос и выполняет перенаправление в сервисы **swa**.
3. **Swa** выполняет проверку учетной записи, если пользователь локальный – осуществляет проверку логина и пароля в хранилище пользовательских аккаунтов, если пользователь из внешнего каталога, то выполняет проверку учетной записи через **adloader**, который отправляет запрос по протоколу LDAP в AD.
4. В случае успешной аутентификации **swa** передает сессионный токен в ответ пользователю.

Сервисы МТА

Примечание

Описание приведено с учетом отправки входящих и исходящих почтовых сообщений с вложениями от внешних и внутренних пользователей в DLP-систему. В схеме под пользователем подразумевается IMAP, SMTP и HTTP внешние и внутренние запросы.

Общая схема сервисов приведена на рисунке ниже:



Описание сервисов МТА приведено в таблице ниже:

Наименование сервиса	Назначение
smtp	Агент пересылки почтовых сообщений от внутренних и внешних аутентифицированных пользователей
relay	Сервис пересылки почтовых сообщений без авторизации для внешнего и внутреннего трафика
mx	Агент пересылки сообщений от внешних неаутентифицированных пользователей
dlp	DLP-система для обработки внешнего/внутреннего почтового трафика

Сервис mailapi

Основной сервис для отправки и получения почтовых сообщений по протоколу HTTP для авторизованных пользователей.

Сервис **mailapi** взаимодействует с :

- сервисом **mescalito** — для получения писем из хранилища;
- с агентом пересылки почтовых сообщений по gRPC — для отправки почтовых сообщений на внешний почтовый сервер и отправки писем между внутренними сотрудниками.

Сервис imapd

Основной сервис для получения почтовых сообщений по протоколу IMAP для авторизованных пользователей.

Сервис **imapd** взаимодействует с :

- прокси сервисом **pub-imap** — для доставки почтовых сообщений по протоколу IMAP в почтовый клиент;
- сервисом **mescalito** — для получения писем для отправки по IMAP.

Сервисы транспортных правил

Сервисы транспортных правил отвечают за управление потоками входящих и исходящих писем (проверка доменов получателя, проверка темы и текста письма, проверка размера вложения и др.).

Описание сервисов транспортных правил приведено в таблице ниже:

Наименование сервиса	Назначение
dlp	Обработчик писем системы транспортных правил
dlp-pg	База данных системы транспортных правил
dlp-api	API системы транспортных правил

Сервисы резервного копирования

Резервное копирование в продукте Почта VK WorkSpace выполняется с помощью **bmwclient** — это клиент для выполнения запросов к gRPC API. Основные функции клиента: резервное копирование и восстановление почтового ящика, облака, календарей, профилей пользователей и адресных книг.

Описание сервисов резервного копирования приведено в таблице ниже:

Наименование сервиса	Назначение
bmw	Сервис бэкапирования данных различных компонентов решения
bmw-tnt	База данных сервиса резервного копирования

Сервисы миграции

Миграция почтовых сообщений состоит из двух основных этапов (процессов):

1. Первичная миграция (первичная сборка).
2. Миграция во времени (по расписанию).

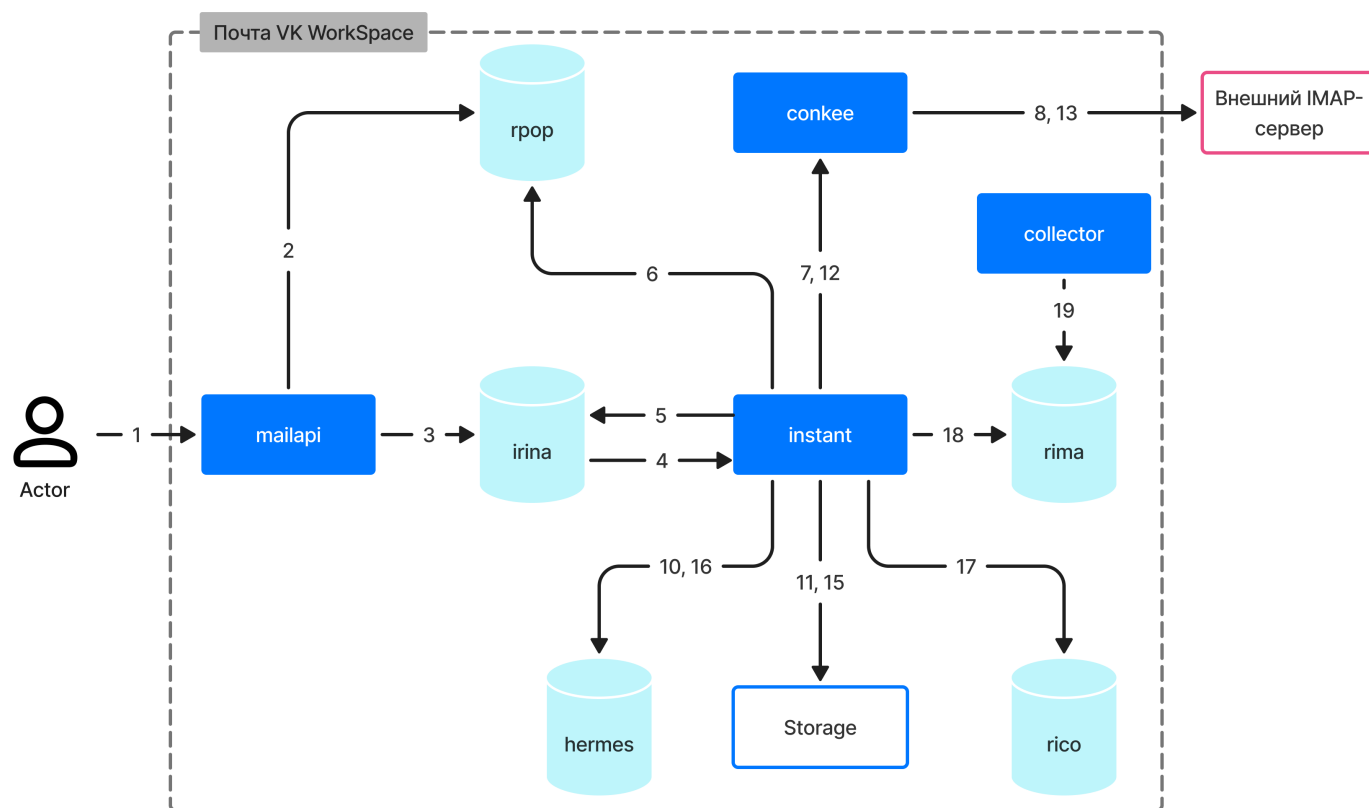
Первая сборка почты

Первая сборка происходит после авторизации администратором нового пользователя или после того, как пользователь самостоятельно прервал сессию, а затем залогинился повторно (тогда пользователь также будет считаться новым). Как только пользователь стал авторизованным, должны быть мгновенно выгружены первые 100 писем в папке Inbox (настраивается в конфигурационном файле).

После создания нового сборщика (его создает прикладной администратор) информация о нем через **mailapi** попадает в **rpop**, а затем отправляется в **irina**. **Irina** привязывает пользователя к конкретному **instant** и отправляет ему оповещение, что необходимо собрать письма нового пользователя.

Instant забирает этого пользователя и проверяет в **rpop**, существует ли такой сборщик. Если сборщик отсутствует, вернется ошибка, и **instant** начнет обрабатывать следующий сборщик. После успешной

проверки сборщика необходимо установить соединение с IMAP через **conkee**. **Instant** (поток 6) отдает **conkee** команду на подключение к IMAP. Когда соединение по IMAP установлено, **instant** проводит первую синхронизацию — выгружает из IMAP список папок, добавляет их абсолютные пути **hermes** и сохраняет в **storage**. Вторым этапом нужно выгрузить первые 100 писем из папки Inbox. **Instant** так же обращается к **conkee** (поток 11), а тот, в свою очередь, выгружает письма из IMAP и передает их в **instant**. Далее **instant** сохраняет письма в **storage** (поток 15) и обновляет в **hermes** downUID (сколько писем скачано). После этого **instant** выставляет время последней сборки в **rico**. При первой сборке **instant** также ставит задачу в **rima** на сборку писем из других папок с помощью **collector** с наивысшим приоритетом (1024).



Описание потоков данных приведено в таблице ниже:

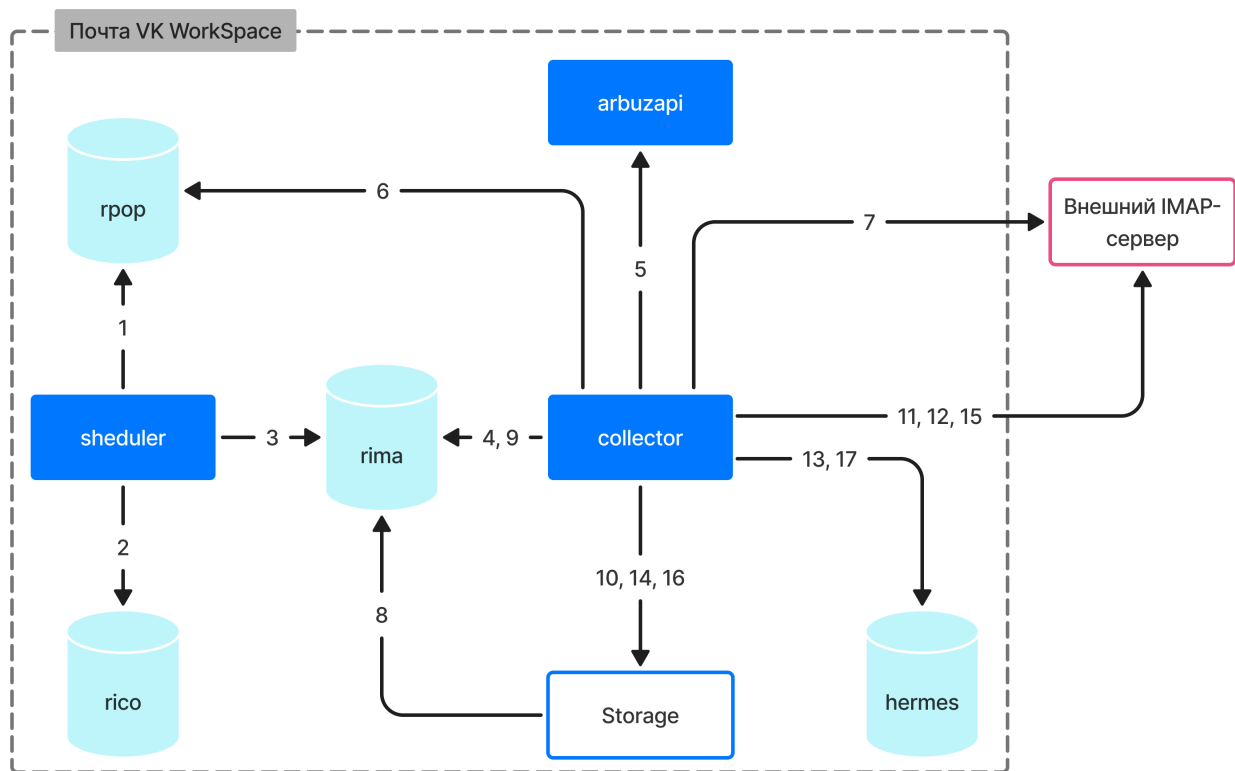
Номер потока	Источник	Описание потока	Получатель
1	Actor	Передает новый сборщик для покладки в БД	mailapi — основной API Почты VK WorkSpace
2	mailapi	Кладет новый сборщик	rpop — БД MySQL, хранящая данные сборщиков
3	mailapi	Отправляет на привязку к мгновенному сборщику	irina — хранит в себе данные о мгновенных сборщиках с номером шарда и о распределении шардов по серверам

Номер потока	Источник	Описание потока	Получатель
4	irina	Оповещает о новом сборщике	instant — демон мгновенной сборки писем
5	instant	Забирает нового пользователя	irina
6	instant	Проверяет сборщик	rpop
7	conkee	Отдает команду на подключение к IMAP	instant
8	conkee	Соединяется с сервером IMAP, забирает список папок (протокол IMAP)	сервер IMAP
9	conkee	Передает список папок	instant
10	instant	Отдает на внесение в БД	hermes — БД, в которой хранится состояние синхронизации по всем папкам всех сборщиков
11	instant	Сохраняет список папок	storage — хранилище данных
12	instant	Запрашивает список писем	conkee — прокси, держит непрерывное соединение с сервером IMAP
13	conkee	Забирает список писем	сервер IMAP
14	conkee	Передает список писем	instant
15	instant	Сохраняет письма	storage
16	instant	Обновляет статус сборки папки	hermes
17	instant	Обновляет тайминги	rico — БД Tarantool, хранит тайминги сборок
18	instant		

Номер потока	Источник	Описание потока	Получатель
		Ставит задачу на сборку по расписанию	rima — очередь задач для сборщиков почты
19	collector	Слушает очередь на сборку по расписанию	rima

Сборка по расписанию

После создания и настройки сборщиков первым начинает свою работу сборка по расписанию, так как в первую очередь необходимо смигрировать основной объем писем.



Описание потоков данных приведено в таблице ниже:

Номер потока	Источник	Описание потока	Получатель
1	scheduler	Забирает список сборщиков	rpop — БД MySQL, хранящая данные сборщиков
2	scheduler	Проверяет сроки сборки по ящикам	rico — БД Tarantool, хранит тайминги сборок
3	scheduler	Ставит задачи на сборку	

Номер потока	Источник	Описание потока	Получатель
			rima — очередь задач для сборщиков почты
4	collector	Забирает задачи из очереди	rima
5	collector	Проверяет наличие пользователя (протокол gRPC).	arbuzapi — gRPC-прокси, изолирующее MySQL базы данных
6	collector	Проверяет наличие сборщиков	rpop — БД MySQL, хранящая данные сборщиков
7	collector	Авторизовывается (протокол IMAP)	сервер IMAP
8	storage	Доносит нотификации со стороны VK WorkSpace.	rima
9	collector	Проверяет, не было ли новых изменений	rima
10	collector	Забирает новые письма из VK WorkSpace	storage — хранилище данных
11	collector	Доносит все изменения на внешний сервер	сервер IMAP
12	collector	Запрашивает изменения в папках с сервера IMAP (протокол IMAP)	сервер IMAP
13	collector	Маппит папки	hermes — БД, в которой хранится состояние синхронизации по всем папкам всех сборщиков
14	collector	Сохраняет изменения в папках	storage
15	collector	Запрашивает письма с сервера IMAP	сервер IMAP

Номер потока	Источник	Описание потока	Получатель
16	collector	Сохраняет новые письма	storage
17	collector	Обновляет маркеры скачивания писем (up/down UID)	hermes

Развернутое описание потоков данных:

1. Раз в 15 минут (периодичность настраивается в конфигурационном файле) **scheduler** обращается к **rpop** за списком ID сборщиков, которые делит на батчи.
2. Далее выполняется проверка таймингов (когда последний раз была сборка по каждому из сборщиков) в **rico**.
3. После проверки таймингов **scheduler** исключает сборщики, по которым не пришло время ставить задачи. Затем ставит задачи в очередь **rima**.
4. **Collector** забирает задачи из очереди **rima** в свою оперативную память.
5. **Collector** запрашивает у **arbuzapi**, существует ли ещё пользователь.
6. **Collector** запрашивает у **rpop**, существует ли нужный сборщик. Если сборщик отсутствует, вернется ошибка, и **collector** обращается за следующим сборщиком в **rima**.
7. Также **collector** забирает у **rpop** пароль для авторизации в IMAP. С паролем **collector** пытается авторизоваться под пользователем по IMAP. Если авторизация прошла неуспешно, задача на сборку возвращается в **rima**. Следующая задача — донести до сервера IMAP изменения, которые произошли на стороне Почты VK WorkSpace.
8. Затем **storage** отправляет все нотификации (установка/снятие флагов, перемещение писем между папками и т.п.) в очередь **rima**.

Примечание

Мы условно ставим поток 8 на донесение изменений из Почты VK WorkSpace, так как это укладывается в логический порядок схемы (перед потоками 9 и 10). Изменения со стороны Почты из **storage** могут добавляться в очередь **rima** в любой момент времени.

9. **Collector** второй раз забирает задачи из **rima** на случай новых изменений.
10. После этого **collector** выполняет запрос в **storage** для сбора новых писем, созданных на стороне Почты VK WorkSpace с момента последней синхронизации.
11. Далее **collector** отправляет на сторону IMAP-сервера все изменения, которые произошли в Почте VK WorkSpace.

 **Примечание**

Если возникают какие-то конфликты, они решаются в пользу IMAP.

12. Далее **collector** выполняет запрос к IMAP-серверу для получения списка папок.
13. После получения списка папок **collector** обращается в **hermes** для маппинга, так как в IMAP папки определяются по абсолютному пути, а в **storage** используются ID. **Collector** обрабатывает список папок в **hermes** и сравнивает с тем, что получил из IMAP. Папки, которые есть в IMAP, но нет в Почте VK WorkSpace, добавляются, а те, которых нет в IMAP, но есть в Почте, удаляются.
14. После маппинга изменений, если они были, сохраняются в **storage**.

 **Примечание**

Если папка новая, **collector** отправляет запрос на получение максимального ID в **hermes** и присвоит новый ID со значением n+1. То же самое произойдет и при сохранении в **storage**.

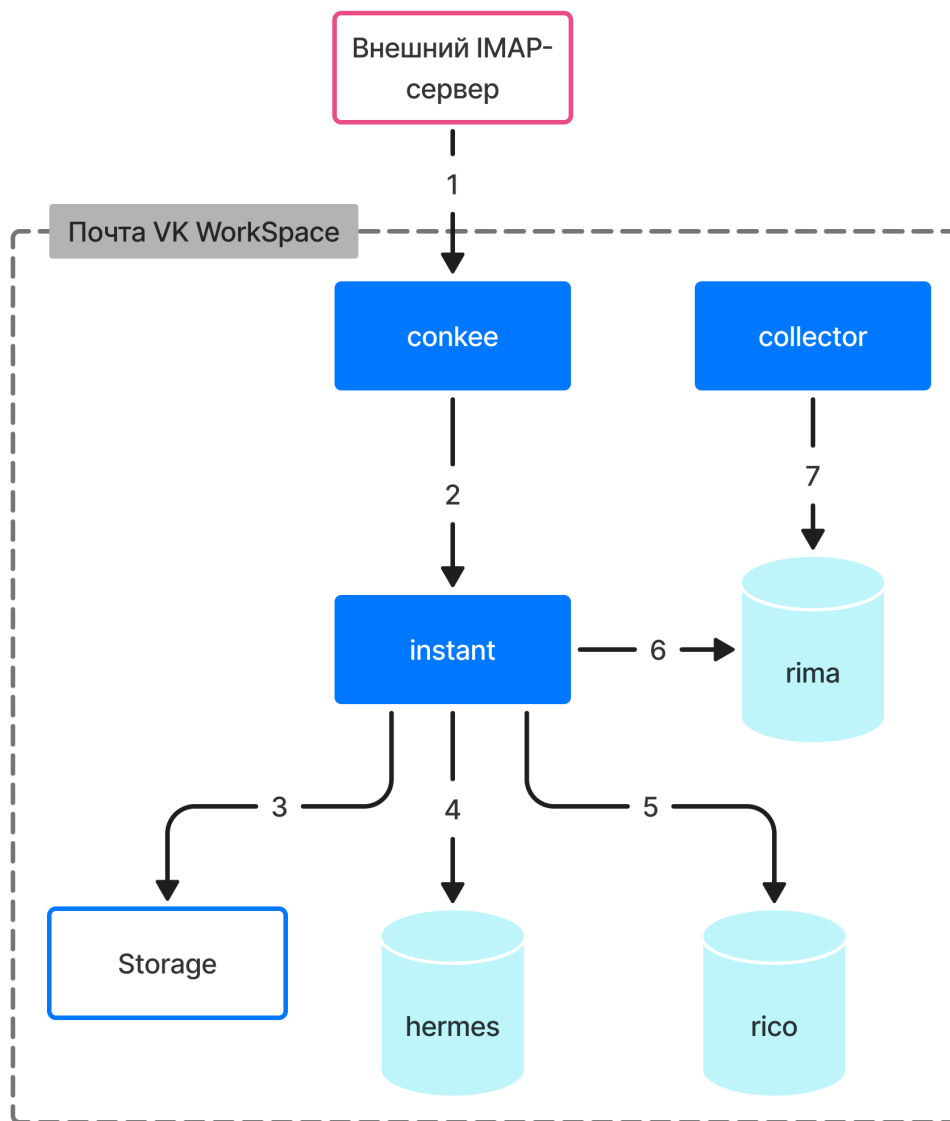
15. После этого **collector** выгружает новые письма по всем папкам с сервера IMAP.
16. Далее **collector** сохраняет их в **storage**.
17. **Collector** отправляет данные для обновления маркеров скачивания писем (up/down UID) в **hermes**.

 **Примечание**

Scheduler работает в бесконечном цикле до момента отключения и полного перехода на мгновенную сборку.

Мгновенная сборка по IDLE

Схема мгновенной сборки по IDLE приведена на рисунке ниже:



Описание потоков данных приведено в таблице ниже:

Номер потока	Источник	Описание потока	Получатель
1	сервер IMAP	Отправляет пуши	conkee — прокси, держит непрерывное соединение с сервером IMAP
2	conkee	Передает новое письмо	instant — демон мгновенной сборки писем
3	instant	Сохраняет письмо	storage — хранилище данных
4	instant	Обновляет состояние сборки для новых писем.	hermes — БД, в которой хранится состояние синхронизации по всем папкам всех сборщиков
5	instant	Обновляет тайминги	

Номер потока	Источник	Описание потока	Получатель
			rico — БД Tarantool, хранит тайминги сборок
6	instant	Ставит задачу на сборку по расписанию	rima — очередь задач для сборщиков почты
7	collector	Слушает очередь на сборку по расписанию	rima

После первой сборки **instant** через **conkee** держит постоянное соединение по IDLE и ждет новых писем. Как только новое письмо от IMAP поступает, **conkee** отправляет его в **instant**, а затем сохраняет в **storage**. Когда письмо сохранено, **instant** обновляет upUID (сколько писем скачано сверху) в **hermes**. Далее **instant** (поток 5) отправляет запрос в **rico** и обновляет время последней сборки из конкретного ящика.

Далее **instant** ставит задачи в **rima** на сборку с помощью **collector**, если:

- не успел что-то скачать;
- прервалось соединение по IMAP;
- существуют иные проблемы доступа к ящику.

Collector забирает задачи из **rima** и скачивает письма в зависимости от приоритета.

Сервис bookster

Сервис для поиска контактов одновременно по доменной и личной адресным книгам.

Сервисы поиска

В поиске реализовано две последовательности действий (процессов):

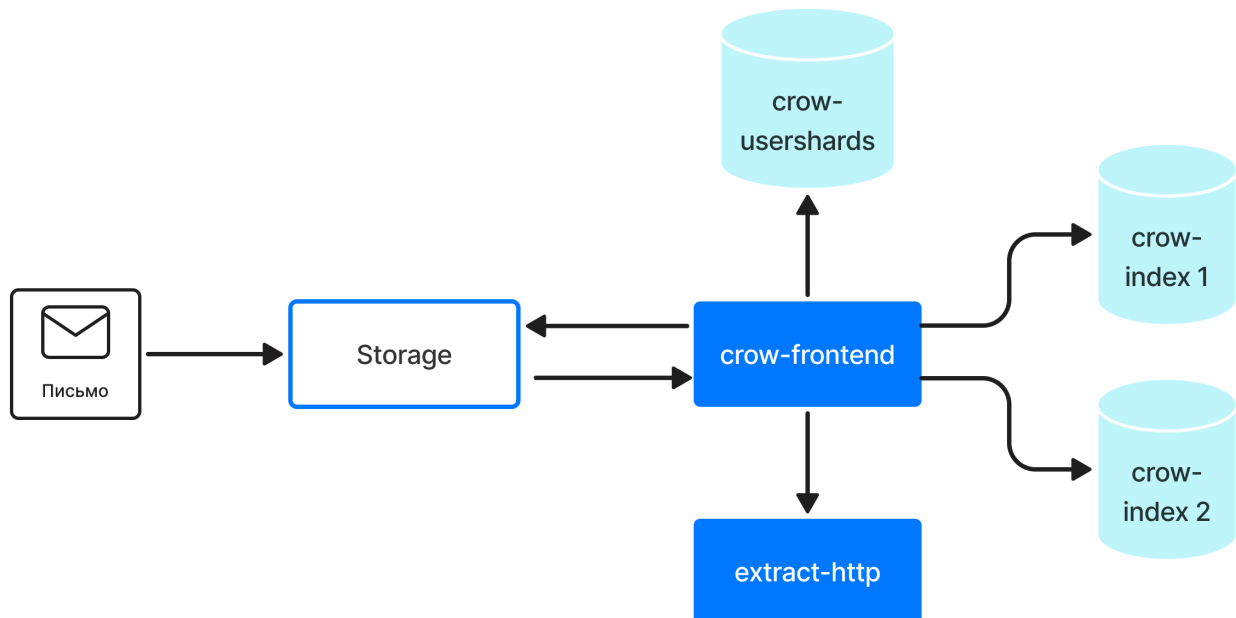
- Индексация новых писем.
- Обработка поисковых запросов.

Индексация новых писем

Описание сервисов, используемых при индексации писем приведено в таблице ниже:

Наименование сервиса	Назначение
storage	Условное наименование нескольких хранилищ данных разного назначения
extract-http	API извлечения текста из документов (например, .docx)
crow-frontend	API поиска в почте
crow-index	Хранилище поисковых индексов
crow-usershards	База данных связей: пользователь, индекс, хранилище индексов

Общая схема сервисов при индексации приведена на рисунке ниже:



При получении нового письма пользователем **storage** формирует и отправляет уведомление о новом письме в **crow-frontend**.

Сервисом **crow-frontend** выполняется следующая последовательность действий:

- получает тело письма со **storage**;
- чистит письмо от лишнего (html, css и др.), оставляет только текст;
- выполняет токенизацию, лемматизацию и т.д.;
- если в письме был приложен документ, то извлекается текст из документа с помощью сервиса **extract-http**;
- преобразует найденные лексемы в формат, пригодный для сохранения в поисковый индекс пользователю;

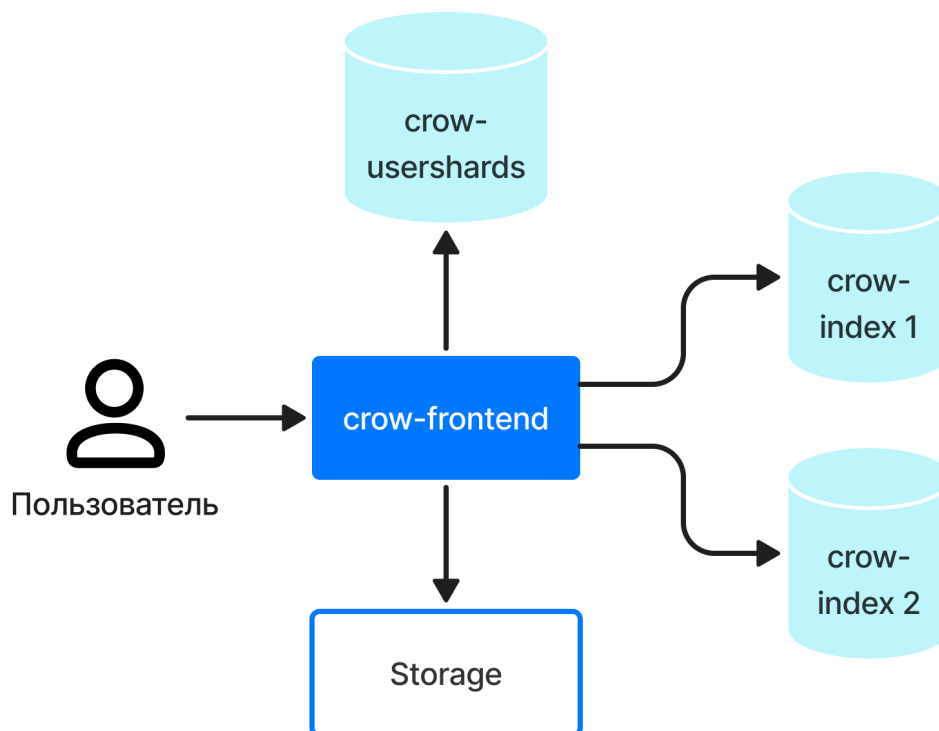
- сохраняет очищенный текст в сервис **storage** (этот текст называется **поисковый сниппет**, в дальнейшем он используется для визуальной подсветки результатов поиска);
- сохраняет выделенные лексемы в оба индекатора.

Примечание

При первом использовании поиска, ему назначается пара индексаторов (**crow-index**), для хранения пользовательских индексов. Соответствие **user_id <==> пара индексаторов (их ip-адреса)** хранится в базе данных **crow-usershards**.

Обработка поисковых запросов

Общая схема сервисов при индексации приведена на рисунке ниже:



При выполнении поисковых запросов пользователем сервисом **crow-frontend** выполняется следующая последовательность действий:

- запрос токенизируется и лемматизируется, как и при индексации письма (в результате получается набор токенов, которые можно искать в индексаторах);
- **crow-frontend** с помощью **crow-usershards** выполняет поиск ip-индексаторов и отправляет запрос в один из них **crow-usershards**;
- если запрос слишком долгий или неудачный, то выполняется такой же запрос в парный индексатор;
- когда ответ получен, то он содержит id писем, которые искал пользователь (т.е. в этих письмах есть запрошенные пользователем фразы);
- **crow-frontend** отправляет запрос в **storage** для получения поисковых сниппетов;
- клиенту отдается ответ, включающий сниппеты, подсветку наиболее релевантных результатов поиска, id писем, подходящих под поисковый запрос.

Сервис streamer

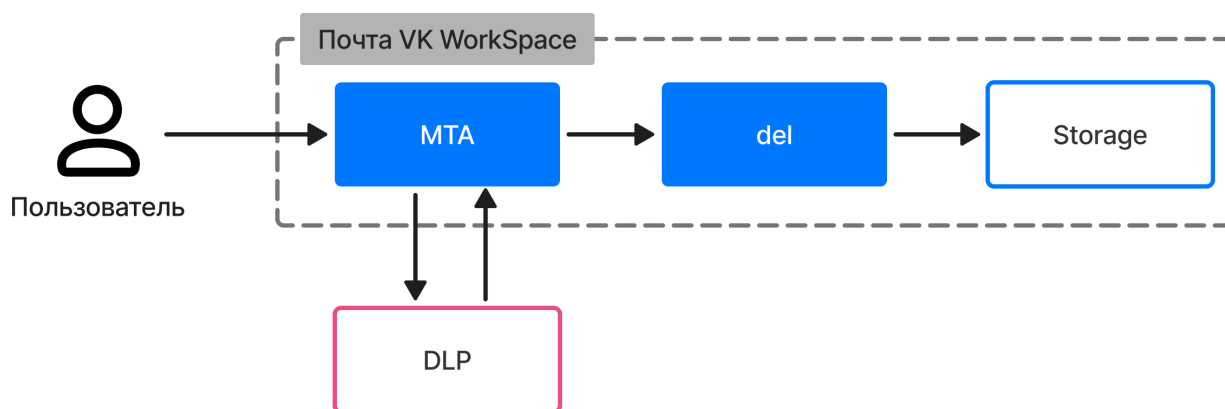
Сервис для загрузки больших вложений в почтовое сообщение. Более подробное описание сервиса приведено в документе [Архитектура VK Диск](#).

Сервис deliveryd

Примечание

Описание приведено с учетом отправки входящих и исходящих почтовых сообщений с вложениями от внешних и внутренних пользователей в DLP-систему.

Общая схема сервисов приведена на рисунке ниже:



После обработки внешнего и внутреннего почтового трафика сервисами MTA (**smtp**, **mx**) выполняется пересылка почтового трафика в DLP-систему. После успешно пройденной проверки в DLP-системе выполняется передача почтового трафика (сервисом **relay**) в сервис **deliveryd** для доставки в почтовое хранилище.

Описание сервисов приведено в таблице ниже:

Наименование сервиса	Назначение
deliveryd	Сервис доставки почтовых сообщений в хранилище
storage	Условное наименование нескольких хранилищ данных разного назначения

Сервисы аудита

Сбор пользовательской статистики осуществляется сервисом **rebecca** на БД PostgreSQL.

Сервис **rebecca** позволяет искать события без привязки к пользователю, то есть просматривать все события из определенной группы.

В рамках продукта Почта VK WorkSpace с помощью сервиса **rebecca** выполняется сбор действий пользователя, которые сопровождаются отправкой событий. События делятся на следующие группы:

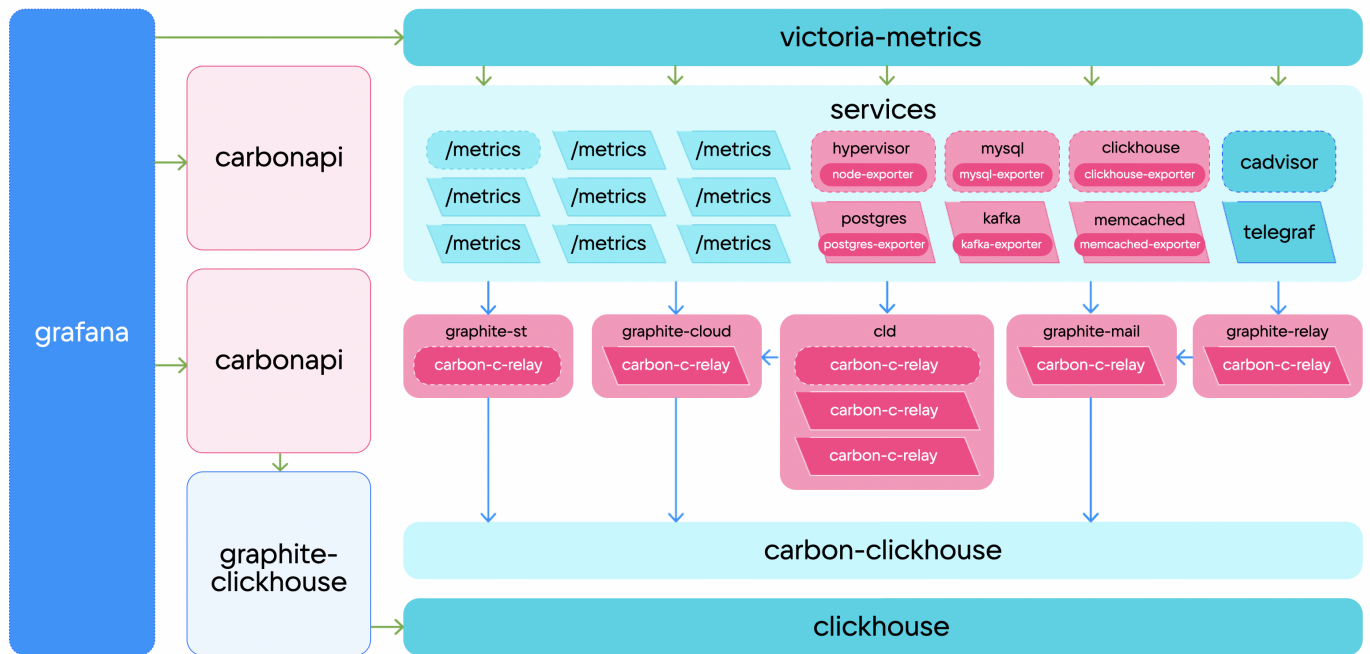
- MailAdmin — действия администратора в Почте.
- Mail — события почты и авторизации.
- Alias — события алиасов почтовых ящиков.

Сервисы мониторинга

В системе Почта VK WorkSpace мониторинг реализован с помощью следующих инструментов:

- Grafana — платформа для визуализации и анализа метрик, поддерживает множество источников данных (Prometheus, VictoriaMetrics, ClickHouse и др.).
- Victoria Metrics — высокопроизводительная СУБД для хранения и обработки временных рядов, альтернатива Prometheus с оптимизированным хранилищем.
- ClickHouse — колоночная аналитическая СУБД, часто используется для хранения и агрегации метрик (особенно в связке с Graphite/Carbon).
- Carbon-clickhouse — прокси-сервер для Graphite, который записывает метрики напрямую в ClickHouse вместо Whisper/Carbon.
- Graphite-clickhouse — Graphite-совместимый бэкенд для хранения метрик в ClickHouse.
- Carbonapi — высокопроизводительный API-сервер для Graphite, поддерживает разные бэкенды (включая ClickHouse).
- Carbon-s-relay — быстрый и легковесный ретранслятор для Carbon (Graphite), поддерживает балансировку и шардирование.
- Telegraf — агент для сбора метрик (CPU, память, сети и др.)
- Cadvisor — инструмент от Google для мониторинга контейнеров (Docker, Kubernetes), собирает данные о ресурсах.
- Node-exporter — экспортер для Prometheus, собирает метрики с Linux-серверов (диски, сеть, CPU и т.д.).

Общая схема сервисов мониторинга приведена ниже:



Сервисы хранения данных

[Системы хранения данных](#) — в документе содержится информация о принципах работы системы хранения данных и описание сервисов хранения данных.

Технические требования

Требования к ресурсам серверного оборудования приведены в документации по установке:

- [Полная установка](#) — в документе описана процедура установки Почты VK WorkSpace в минимальной рабочей конфигурации на одну виртуальную машину.
- [Кластерная установка](#) — в документе описана процедура кластерной установки Почты VK WorkSpace.
- [Тестовая установка](#) — в документе описана тестовая установка Почты VK WorkSpace на одну виртуальную машину.

🕒 24 декабря 2025 г.