

ИНСТРУКЦИЯ ПО ЭКСПЛУАТАЦИИ
программного обеспечения
«Сервис замены ключей»

Оглавление

1. ГЛОССАРИЙ	3
2. ОПИСАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «СЕРВИС ЗАМЕНЫ КЛЮЧЕЙ»..	4
2.1.НАИМЕНОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ЕЕ УСЛОВНЫЕ ОБОЗНАЧЕНИЯ	4
2.2.ФУНКЦИОНИРОВАНИЕ ПО «СЕРВИС ЗАМЕНЫ КЛЮЧЕЙ»	4
3. ЭКСПЛУАТАЦИЯ ПО «СЕРВИС ЗАМЕНЫ КЛЮЧЕЙ».....	4
3.1.НАСТРОЙКА СЕРВИСА	4
3.2.СХЕМА РАБОТЫ СЕРВИСА	5
3.3.ПРОВЕРКА РАБОТОСПОСОБНОСТИ СЕРВИСА.....	6
4. КОНТАКТНАЯ ИНФОРМАЦИЯ.....	7

1. Глоссарий

№	Обозначение или сокращение	Понимание или расшифровка
1.	Хеш, hash	Результат преобразования данных с помощью хеш-функции
2.	Хеширование	Преобразование, производимое хеш-функцией
3.	Хеш-функция	Функция, осуществляющая необратимое преобразование массива входных данных произвольной длины в выходную битовую строку установленной длины, выполняемое определённым алгоритмом
4.	БД	База данных

2. Описание программного обеспечения «Сервис замены ключей»

2.1. Наименование программного обеспечения и ее условные обозначения

Наименование программного обеспечения: «Сервис замены ключей»

Условное наименование программного обеспечения: «Сервис»

Назначение Сервиса: ПО «Сервис замены ключей» представляет собой прокси-сервер (промежуточный сервис) при передаче пакетов данных с информацией о клиентах и их действиях через API от Источника данных в Хранилище данных в прямом и обратном направлении и используется для обезличивания данных.

2.2. Функционирование ПО «Сервис замены ключей»

Функционал ПО «Сервис замены ключей» заключается в следующем:

- Сервис производит замену Ключей А (HASH1, выделенные из получаемых пакетов данных) на Ключи Б (HASH2) из собственной Базы данных (БД);
- выполняет обратное преобразование: из Ключей Б определяет Ключи А;
- отправляет письма для проверки функционала преобразования ключей.

Принцип функционирования ПО «Сервис замены ключей»:

Пользователь формирует пакеты данных и определяет клиентов, которым требуется направить такие пакеты, присваивая им хешированный идентификатор, и в автоматическом режиме передает файл по API в формате JSON, который содержит hash-идентификатор и набор других необходимых полей, не сохраняющихся в Сервисе. Сервис внутри себя формирует реестр хэшей, где к каждому уникальному hash-идентификатору присваивается в соответствие новый hash-идентификатор. Сервис заменяет полученный идентификатор на идентификатор из реестра и возвращает обезличенные данные клиентам. При этом ни у одного лица не остается полного набора данных для идентификации, за счет чего передача пакетов данных указанным способом является безопасной.

3. Эксплуатация ПО «Сервис замены ключей»

3.1. Настройка Сервиса

Программное обеспечение «Сервис замены ключей» является интернет-сервисом. Пользователь не производит настройку Сервиса самостоятельно, подключаясь к уже настроенному и готовому к работе Сервису. Доступ к программному обеспечению можно получить только через администратора ресурса ООО «Лоция», обратившись в техническую поддержку.

Для использования ПО «Сервис замены ключей» пользователю необходимо настроить сетевое взаимодействие. Для этого требуется наличие системы, которая способна отправлять данные в формате JSON по API в ПО «Сервис замены ключей», и получать от него ответ формате JSON с обезличенными идентификаторами. Отправляемые данные должны иметь hash идентификаторы клиентов пользователя.

Хеширование идентификаторов может быть реализовано с помощью БД, например такая реализация возможна в БД Oracle, а для сетевого взаимодействия подойдут готовые решения, такие как Platformeco или собственные системы компании-пользователя.

3.2. Схема работы Сервиса

1. Доступ к сервису осуществляется с помощью логина и пароля.
2. Для начала работы с Сервисом необходимо сгенерировать запрос, содержащий HASH идентификатор в первом столбце и передаваемые данные в остальных столбцах. Данные передаются в формате JSON.
3. Запрос передается на один из адресов, в зависимости от пользователя:
по внешним адресам:
Пример: romashka.loodsen.ru
или по внутренним адресам:
Пример: romashka.inner.loodsen.ru
4. После получения запроса Сервис производит подмену HASH1 → HASH2 или наоборот.
5. После этого формирует новый запрос для передачи преобразованной информации в хранилище данных. Обезличенная таким образом информация используется пользователем по назначению.

Диаграмма компонентов Сервиса приведена на рисунке «Диаграмма компонентов»:

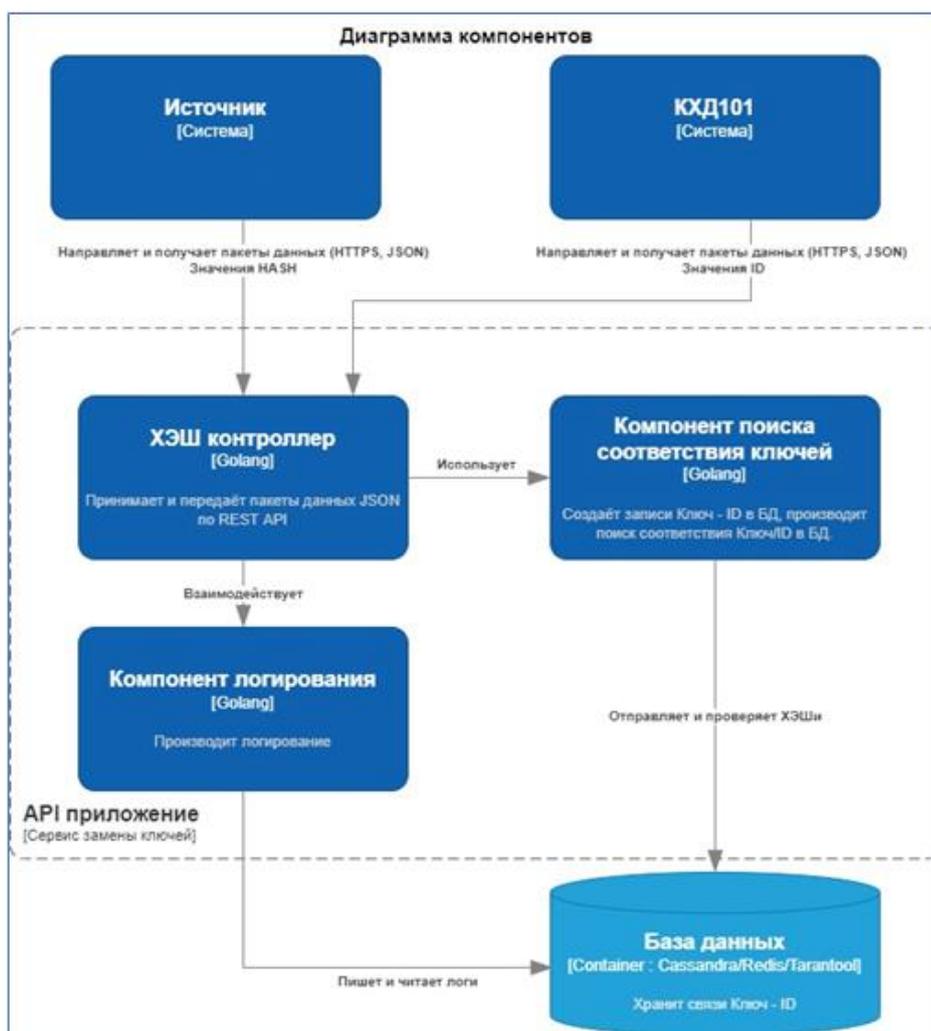


Рисунок «Диаграмма компонентов»

Алгоритм выполнения прямого порядка передачи данных «Источник – Сервис – КХД», замена Ключа А (HASH1) на Ключ Б (HASH2) приведён в Таблице 1.

Таблица 1. Прямой порядок передачи данных

№	Ответственный	Описание
1.	Источник	Передаёт данные в API Сервиса.
2.	ХЭШ контроллер	Принимает в API Сервиса пакет данных от Источника.
3.	Компонент поиска соответствия Ключей	Выбирает из необходимых полей значения Ключей
4.	Компонент поиска соответствия Ключей	Проверяет, если в таблицах БД Сервиса такие Ключи.
5.	Компонент поиска соответствия Ключей	Если нет, то создает по ключу новую запись, где сохраняет значения «Ключа А» от источника с новыми уникальными значениями «Ключа Б».
6.	Компонент поиска соответствия Ключей	Если соответствие находится, то получает «Ключ Б» по «Ключу А».
7.	Компонент поиска соответствия Ключей	Заменяет в пакете данных «Ключ А» на «Ключ Б».
8.	ХЭШ контроллер	Вызывает метод на стороне системы получателя и передает пакет с новыми ключами.

Алгоритм выполнения обратного порядка передачи данных «КХД – Сервис – Источник», замена Ключа Б (HASH2) на Ключ А (HASH2) приведён в Таблице 2.

Таблица 2. Обратный порядок передачи данных

№	Ответственный	Описание
1.	ХЭШ контроллер	Вызывает метод на стороне КХД и получает пакет данных с Ключами Б
2.	Компонент поиска соответствия Ключей	Выбирает из необходимых полей значения Ключей
3.	Компонент поиска соответствия Ключей	Заменяет в пакете данных «Ключ Б» на «Ключ А».
4.	Источник	Вызывает метод на стороне Сервиса и получает пакет с Ключами А.

3.3. Проверка работоспособности Сервиса

Проверка работоспособности Сервиса выполняется с помощью дополнительного функционала – валидации данных.

Функционал:

1. На технический адрес (например: 000bot@loodsen.ru) пользователь отправляет файл CSV.
Первый столбец – HASH1 или HASH2.
2. Сервис автоматически просматривает технический адрес (например: 000bot@loodsen.ru), 1 раз в 5 минут.
3. Увидев в почте файл, Сервис проверяет метки в теме письма.

Формат темы: [метка пользователя] <направление> тема письма

Направление преобразования может быть прямым и обратным:

- прямое направление <forward>;
- обратное направление <reverse>.

Например, чтобы Сервис выполнил обратное преобразование, необходимо указать:

[crossales] <reverse> тема письма

Разделение пробелами текста обязательно.
Обработанные данные придут на адрес, закреплённый за меткой пользователя.

Чтобы Сервис обработал данные по прямому алгоритму (прямое преобразование), необходимо указать:

[crossales] <forward> тема письма

Разделение пробелами текста обязательно.
Обработанные данные придут на адрес, закреплённый за меткой пользователя.

Сервис работоспособен, если после отправки письма с данными, получаем ответ с письмом, в котором в файле HASH1 заменен на HASH2 или наоборот.

4. Контактная информация

Техническая поддержка:

Пн-Пт, 9:00-18:00 по МСК

Тел.: +7(495) 730-23-20

Адрес электронной почты: FilippovMV@loodsen.ru, RybalchenkoAA@loodsen.ru