**Методика интеграции с**

**Маршрутизатором АИС МВ**

Содержание

[Введение 5](#_Toc77004200)

[1. Предварительные требования для интеграции с API «Маршрутизатора» 6](#_Toc77004201)

[1.1 Подписание на API АПК «Платформа» 6](#_Toc77004202)

[1.2 Авторизация запросов при обращении к API 6](#_Toc77004203)

[1.3 Пример запроса получения токена авторизации 7](#_Toc77004204)

[1.4 Правила формирования запроса пользователем 8](#_Toc77004205)

[1.5 Пример последовательности действий вызова API АПК «Платформа» 8](#_Toc77004206)

[1.6 Формирование URL для обращения к API Маршрутизатора 9](#_Toc77004207)

[2. Общий процесс обработки пакета в маршрутизаторе 11](#_Toc77004208)

[3. PackageReceiver - Процесс загрузки пакета в Маршрутизатор 12](#_Toc77004209)

[3.1 Описание модели данных и API сервиса PackageReceiver 13](#_Toc77004210)

[3.1.1 Создание пакета 14](#_Toc77004211)

[3.1.2 Отправка пакета в маршрутизатор на обработку 16](#_Toc77004212)

[3.1.3 Получить еще не отправленный пакет 17](#_Toc77004213)

[3.1.4 Обновить информацию в пакете 19](#_Toc77004214)

[3.1.5 Добавить файл к пакету 21](#_Toc77004215)

[3.1.6 Удалить файл из пакета 23](#_Toc77004216)

[3.1.7 Добавить или изменить ЭЦП подписи у файла в пакете 24](#_Toc77004217)

[3.1.8 Удалить ЭЦП у файла 26](#_Toc77004218)

[3.1.9 Удалить не отправленный на обработку пакет 27](#_Toc77004219)

[4. PackageSender - Процесс получения пакета из Маршрутизатора 28](#_Toc77004220)

[4.1 Описание модели данных и API сервиса PackageSender 29](#_Toc77004221)

[4.2.1 Получение списка пакетов 30](#_Toc77004222)

[4.2.2 Получение вложения относящегося к пакету 32](#_Toc77004223)

[4.2.3 Подтверждение получения списка пакетов 34](#_Toc77004224)

[4.2.4 Подтверждение получения одного пакета 35](#_Toc77004225)

[4.3 Квитанции 36](#_Toc77004226)

[5. Взаимодействие с СМДО 37](#_Toc77004227)

# ****Введение****

АПК «Платформа» является основой для развертывания переработанного функционала ОАИС и СМДО в составе разрабатываемых приложений АИС МВ, а также для расширения функционала ОАИС и АИС МВ.

Взаимодействие между системами электронного документооборота, должно осуществляться через подсистему «Маршрутизатор электронных документов» (далее – «Маршрутизатор») АПК «Платформа».

Подсистема Маршрутизатор обеспечивает сервисные функции для организации обмена электронными сообщениями между приложениями АИС МВ. Электронное сообщение представляет собой структурированные данные, которые могут содержать один или несколько файлов. Структура всех передаваемых данных должна быть определена в Подсистеме форматов обмена.

Данный документ описывает методику интеграции приложений с подсистемой Маршрутизатор электронных документов.

Функциями Маршрутизатора являются прием электронных документов от одних абонентов и передача их другим абонентам по заранее оговоренному формату обмена сообщениями. Хранение схем сообщений и форматов обмена является задачей подсистемы «Динамические форматы обмена», функциональные возможности которой не описываются в данном документе.

Подсистема Маршрутизатор представляет собой набор множества сервисов, обеспечивающих логику приема, обработки и передачи сообщений. Для интеграции с Маршрутизатором необходимо взаимодействие с API двух сервисов PackageReceiver (прием сообщений) и PackageSender (передача сообщений), которые будут описаны в данному документе.

Для передачи электронных документов в рамках работы с Маршрутизатором принято оперировать понятием пакет.

*Пакет маршрутизатора* – это электронное сообщение Json формата, состоящее из полезной информации (body) и дополнительных атрибутов, представленных парами ключ-значение.

Пакеты, загружаемые в Маршрутизатор через PackageReceiver и полученные через PackageSender имеют некоторые отличия. Описание пакетов представлено в соответствующих разделах.

# Предварительные требования для интеграции с API «Маршрутизатора»

Для получения доступа к API сервисам Маршрутизатора интегрируемые информационные системы должны соблюсти следующие требования:

- зарегистрироваться в АПК «Платформа»;

- подписаться на необходимые API;

- получить token доступа для авторизации запросов к API.

Более подробная информация **содержится в документе** описания технических требований для подключения ИС к АПК «Платформа».

## 1.1 Подписание на API АПК «Платформа»

Подписание на API происходит самостоятельно из личного кабинета пользователя, пункт меню «Управление API/Приложения» или администратором АПК «Платформа».

## 1.2 Авторизация запросов при обращении к API

Доступные API в АПК «Платформа» опубликованы в WSO2 API Manager. Для авторизации запросов к API, используется Bearer токен (далее токен авторизации). Токен авторизации выдается потребителю на срок 3600 секунд (1 час), путем выполнения служебного запроса к шлюзу WSO2 API Manager, адрес шлюза WSO2 API Manager <http://gw.nces.by/api/token>.

Запрос выполняется с использованием следующих параметров:

* тип запроса POST;
* обязательный постоянный query-параметр "scope=application&grant\_type=client\_credentials";
* обязательный заголовок **Authorization**, метод Basic, передаваемое значение – закодированная в **base64** строка вида client\_key:client\_secret (логин:ключ соответственно), оба параметра назначаются автоматически при создании Приложения для (Г)ИР(С), и доступны на странице «Разработчику/Приложения» в секции «Безопасность»;
* обязательный заголовок -H "Content-Type: application/x-www-form-urlencoded".

## 1.3 Пример запроса получения токена авторизации

В АПК «Платформа» на странице зарегистрированного «Приложение 1», в секции «Безопасность», будут видны следующие значения:

Логин = yH7AonImAT\_oVrtwYOxjTfFos0ka

Ключ = vQpjxOa6w3YK6jHLxRuHCrZHjVU

Путем конвертирования строки (без учета кавычек) «yH7AonImAT\_oVrtwYOxjTfFos0ka:vQpjxOa6w3YK6jHLxRuHCrZHjVUa» в base64 получится значение «eUg3QW9uSW1BVF9vVnJ0d1lPeGpUZkZvczBrYTp2UXBqeE9hNnczWUs2akhMeFJ1SENyWkhqVlVh» для базовой аутентификации используемой в запросе.

Запрос на получение токена выглядит следующим образом:

curl -k -d "scope=application&grant\_type=client\_credentials" -H "Authorization: Basic eUg3QW9uSW1BVF9vVnJ0d1lPeGpUZkZvczBrYTp2UXBqeE9hNnczWUs2akhMeFJ1SENyWkhqVlVh" -H "Content-Type: application/x-www-form-urlencoded" https://gw.nces.by/api/token

В ответ, на запрос будет получен json объект:

{

"access\_token":"39a950ba-85c1-3999-9407-2e5942f91284",

"scope":"application",

"token\_type":"Bearer",

"expires\_in":3600

}

В запросе возвращается следующий набор значений:

* access\_token – значение токена доступа к «Приложению 1», с этим значением будут доступны вызовы всех API, входящих в состав «Приложения 1»;
* scope – область действия токена;
* token\_type – тип токена используемого в заголовке Authorization (тип авторизации) при вызове API;
* expires\_in – время в секундах, оставшееся до истечения срока действия выданного токена.

Механизмом обновления токена авторизации для доступа к вызовам API является вызов ранее описанного служебного запроса.

Разработчики внешних по отношению к АПК Платформа сервисов (Г)ИС(Р) обязаны самостоятельно реализовать процесс получения (обновления) токена авторизации – путем обработки значения параметра «expires\_in», либо обработки стандартного HTTP ответа 401 – неавторизованный запрос. Второй вариант является более приоритетным, т.к. создает меньшую нагрузку на шлюз WSO2 API Manager.

## 1.4 Правила формирования запроса пользователем

Передача запроса Пользователя в (Г)ИС(Р) Поставщика информации осуществляется посредством сообщений протокола HTTP с типом содержимого application/json.

Структурно HTTP-сообщение состоит из следующих сложений:

* заголовок;
* параметры метода веб-сервиса;
* тело сообщения, содержащего запрос/ответ с перечнем атрибутов для определенного веб-сервиса в формате JSON, значения атрибутов.

Заголовок должен содержать:

* авторизационный токен доступа (пример: Authorization: Bearer <access\_token>);
* тип содержимого (пример: Content-Type: application/json);
* иные параметры, определяемые Поставщиком информации.

Получаемый у Оператора ОАИС токен доступа имеет формат GUID (например: 6f9619ff-8b86-d011-b42d-00cf4fc964ff).

Согласно спецификации протокола Oauth 2.0 полученный токен доступа необходимо передавать в заголовке запроса. При успешной проверке токена доступа в АПК, Пользователь проходит авторизацию и имеет право использовать доступный ему перечень веб-сервисов. В случае ошибочной авторизации Пользователю будет отказано в доступе.

Пример передачи токена доступа в заголовке запроса:

curl -k -X GET " https://gw.nces.by/api/context/1.0.0/method?parameter=value" H "accept: application/json" -H "Authorization: Bearer 6f9619ff-8b86-d011-b42d00cf4fc964ff"

## 1.5 Пример последовательности действий вызова API АПК «Платформа»

Общая последовательность действий для вызова API АПК «Платформа»:

* Запросить токен авторизации для приложения;
* Используя полученный токен сформировать запрос к API согласно его описанию.

Например:

1. Метод запроса – соответствует описанию опубликованного API сервиса АПК Платформа;

Например, **POST**

1. Обязательный заголовок запроса Authorization;

**Authorization:**

1. Метод авторизации Bearer token;

"**Authorization:** **Bearer 39a950ba-85c1-3999-9407-2e5942f91284"**

1. Url шлюза WSO2 API Manager <https://gw.nces.by/api>;

**https://gw.nces.by/api**/

1. Контекcт вызова API берется из поля контекст в описании API опубликованного сервиса АПК Платформа;

**https://gw.nces.by/api/context/**

1. Версия API берется из поля версия в описании API опубликованного сервиса АПК Платформа;

**https://gw.nces.by/api/context/1.0.0/**

1. Метод - метод API опубликованного сервиса АПК Платформа.

**https://gw.nces.by/api/context/1.0.0/method**

Результатом должен стать запрос вида:

curl -X POST https://gw.nces.by/api/context/1.0.0/method -H "Authorization: Bearer 39a950ba-85c1-3999-9407-2e5942f91284" -H "Content-Type: application/json" -d "{"key": "value"}"

## 1.6 Формирование URL для обращения к API Маршрутизатора

Доступ к опубликованным API в АПК «Платформа» осуществляться по URL в виде:

https://gw.nces.by/api/context/version/method

* https – сетевой протокол, используемый для обращения к АПК «Платформа»;
* context – наименование API;
* version – версия API;
* method – метод веб-сервиса.

В свою очередь для взаимодействия с API маршрутизатора **method** состоит из:

* user\_type – тип абонента определен в API при подписании в APIM, может принимать значения (app, user, org);
* header\_id – заголовок схемы сообщения (комбинированный ключ, который включает тип схемы и номер её версии) (пример SMDO~1.0.0);
* path – путь запроса (пример запроса /smdo~1.0.0/package).

Полный путь выглядит как:

https://gw.nces.by/api/context/version/**user\_type/header\_id/path**

Примеры URL:

* [https://gw.nces.by/api/receiver/1.0.0/org/smdo~1.0.0/package/{packageId}/send](https://gw.nces.by/api/receiver/1.0.0/org/smdo~1.0.0/package/%7bpackageId%7d/send)
* [https://gw.nces.by/api/sender/1.0.0/org/smdo~1.0.0/package/{id}/attach/{attachId}](https://gw.nces.by/api/sender/1.0.0/org/smdo~1.0.0/package/%7bid%7d/attach/%7battachId%7d)

[**​**](http://172.16.9.47:31117/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/downloadFile)

Часть URL **user\_type** используется для обозначения типа абонента, от которого планируется получение/отправка пакетов через маршрутизатор. В зависимости от передаваемого типа Маршрутизатор понимает каким из идентификаторов оперировать для обеспечения логики обмена сообщениями.

Для идентификации абонента используется один из типов:

user – используется идентификатор пользователя из токена;

app – идентификатор приложения из токена;

org – используется идентификатор организации из токена.

**Важно**: в рамках обмена сообщениями по формату АИСМВ(СМДО) используется тип **org**.

Конечный URL зависит от публикации\регистрации\конфигурации сервисов в АПК Платформа.

# Общий процесс обработки пакета в маршрутизаторе

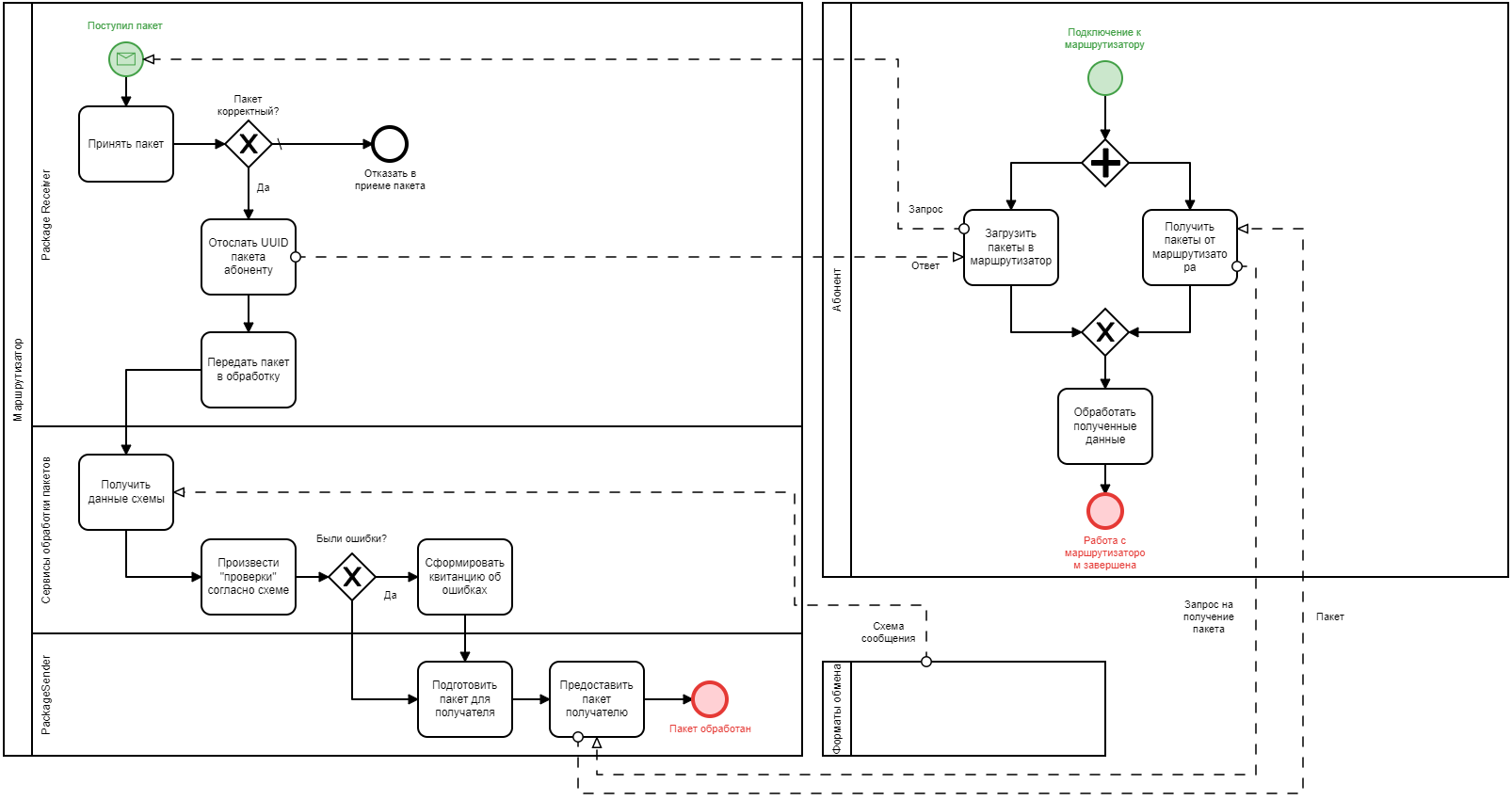


Рисунок 1 – Общий процесс работы с маршрутизатором

1. Процесс начинается с загрузки пакета абонентом в Маршрутизатор через PackageReceiver (см. раздел PackageReceiver), если пакет принимается в обработку, то он передается далее в «логический блок» маршрутизатора. В противном случае абонент отправитель получит отказ о принятии пакета.

2. «Логический блок» маршрутизатора состоит из множества сервисов, которые производят различные операции с пакетом. Например: антивирусную проверку, проверку тела пакета на соответствие формату обмена, размещение данных о пакете в модуль статистики, проверка ЭЦП и тд. Какую обработку производить, маршрутизатор узнает из подсистемы «Динамических форматов обмена» обратившись к конкретной схеме сообщения по заголовку headerId из обрабатываемого пакета (см. раздел PackageReceiver). Если обработка в «логическом блоке» прошла успешно, пакет передается дальше в сервис PackageSender (см. раздел PackageSender), используя Rest API которого абоненты-потребители смогут получить пакеты, предназначенные им. Если обработка пакета в «логическом блоке» произошла с ошибкой, то может быть сформирована квитанция об ошибке (см. раздел Квитанции), которая так же будет передана в сервис PackageSender.

3. Обработка пакета заканчивается тем, что абонент потребитель обращается по Rest API к сервису PackageSender и забирает адресованные ему пакеты.

Для лучшего понимания на рисунке 1 представлен упрощенный общий процесс работы с маршрутизатором.

# PackageReceiver - Процесс загрузки пакета в Маршрутизатор

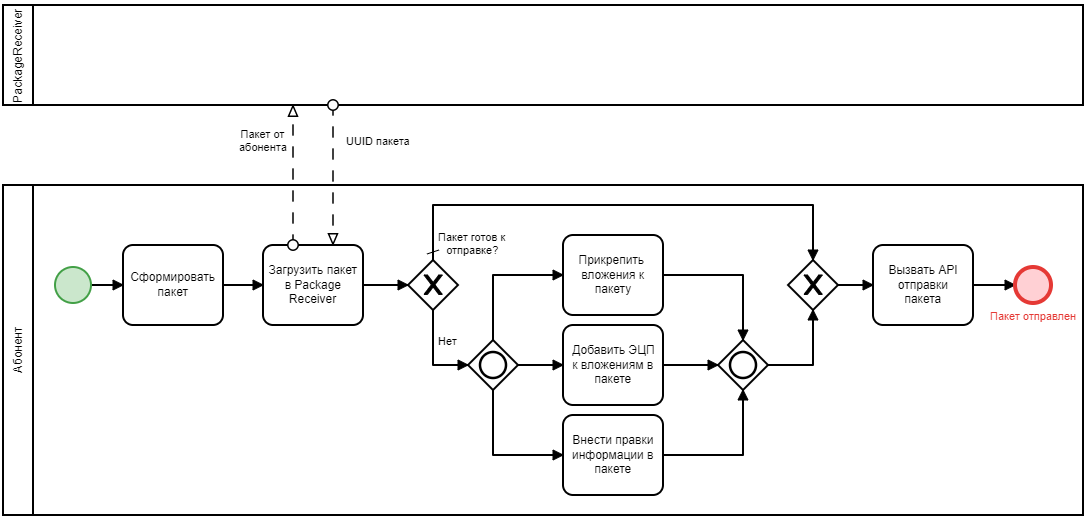


Рисунок 2 – Загрузка пакета в Маршрутизатор через PackageReceiver

Процесс загрузки пакета в Маршрутизатор происходит в несколько этапов. Каждый этап представляет собой обращение к Rest API сервиса PackageReceiver. Этапы загрузки пакета в Маршрутизатор можно представить следующей последовательностью действий:

1) Создать и загрузить пакет, в ответ на который получить UUID пакета **POST**[**​/{userType}​/{headerId}​/package**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/create)

- Если требуется загрузить вложения к пакету, то необходимо по ранее полученному UUID сформировать multi-part запроc и получить id вложения **POST**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/uploadAttach)

- При необходимости сопроводить вложение ЭЦП подписью вызвать **PUT**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}​/signs**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/updateAttachmentSigns)

2) Отправить пакет. Пакет **считается отправленным** только после вызова API

**GET**[**​/{userType}​/{headerId}​/package​/{packageId}​/send**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/send)

Детальное описание API сервиса PackageReceiver представлена в соответствующем разделе данного документа.

Если по каким-либо причинам необходимо изменить пакет, удалить вложения или вообще отменить дальнейшую работу с пакетом предусмотрены дополнительные Rest API:

- Изменить информацию о пакете

**PUT**[**​/{userType}​/{headerId}​/package**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/update)

- Удалить ЭЦП у файла

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}​/signs**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deleteAttach)

- Удалить файл

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deleteAttachment)

- Удалить пакет

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deletePackageById)

**Важно** любые действия по модификации пакета и его содержимого возможны только до вызова **GET**[**​/{userType}​/{headerId}​/package​/{packageId}​/send**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/send)

## 3.1 Описание модели данных и API сервиса PackageReceiver

Описание атрибутов модели данных для сервиса PackageReceiver представлено в таблице 3.1.

Таблица 3.1 - описание модели данных сервиса PackageReceiver

|  |  |  |  |
| --- | --- | --- | --- |
| Реквизит | Обязательный | Описание | Формат |
| parentId | - | Идентификатор родительского пакета | UUID rfc4122 |
| dateSent | + | Дата формирования пакета | ISO 8601 |
| body | - | Тело пакета | Произвольный JSON объект |
| to/idAbonent | + | Идентификатор абонента получателя из справочника абонентов | Строка |
| to/name | - | Имя абонента | Строка |
| packageId | - | Идентификатор пакета | UUID rfc4122 |
| headerId | + | Заголовок схемы | Строка |
| attachments/id | - | Идентификатор файла | UUID rfc4122 |
| attachments/size | - | Размер файла в байтах | Целое число |
| attachments/name | - | Имя файла | Строка |
| attachments/sign/keyId | - | Идентификатор открытого ключа, подписавшего | Строка |
| attachments/sign/ signer | - | Имя подписавшего | Строка |
| attachments/sign/ signtime | - | Дата/время подписи | ISO 8601 |
| attachments/sign/ operationType | - | Тип операции подписания | Строка |
| attachments/sign/ origSignature | - | Значение ЭЦП в исходной системе (если выполнялось заверение третьей стороной) | Строка |
| additionalProperties | - | Дополнительные параметры | Произвольный JSON объект |
| attachments/sign/data | - | ЭЦП подпись в base64 | Строка |

## 3.1.1 Создание пакета

Для создания пакета используется **POST**[**​/{userType}​/{headerId}​/package**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/create)**.** Следует понимать, что создание пакета не означает его отправку на обработку в маршрутизатор, следовательно абоненты-получатели не получат его. Для того что бы отправить пакет после создания, необходимо вызвать соответствующий Rest API (см. раздел Отправка пакета в маршрутизатор на обработку).

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | application/json |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 201 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 500 | Ошибка сервера |

Тело запроса должно быть сформировано в формате **application/json**, наполнение элемента body определяется форматом обмена, шаблон тела запроса выглядит следующим образом:

{

"packageId": ""

"to": [

{

"idAbonent": "string",

"name": "string"

}

],

"parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",

"dateSent": "2021-01-23T17:04:23.068Z",

"body": {

}

}

После выполнения запроса можно ожидать следующий шаблон ответа:

|  |
| --- |
| 201 Успешная операция  {  "packageId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "to": [  {  "idAbonent": "string",  "name": "string"  }  ],  "headerId": "string",  "parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "dateSent": "2021-01-23T15:10:55.335Z",  "body": {...},  "attachments": [  {  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "size": 0,  "name": "string",  "sign": [  {  "keyId": "string",  "signer": "string",  "signtime": "2021-01-23T15:10:55.335Z",  "operationType": "string",  "origSignature": "string",  "data": "string"  }  ]  }  ]  } |

Пример запроса создания пакета:

curl --location --request POST 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package' \

--header 'Authorization: Bearer f40d7fc6-fe05-3904-9dce-7bbd4f35f939' \

--header 'Content-Type: application/json' \

--data-raw '{

"to": [ {

"idAbonent": "7d7eff59-32eb-4565-8a8e-73492a9e33a5",

"name": "Общество с ограниченной ответственностью Борисовский завод двигателей Джили"

} ],

"parentId": null,

"dateSent": "2020-03-20T10:39:35Z",

"body": {

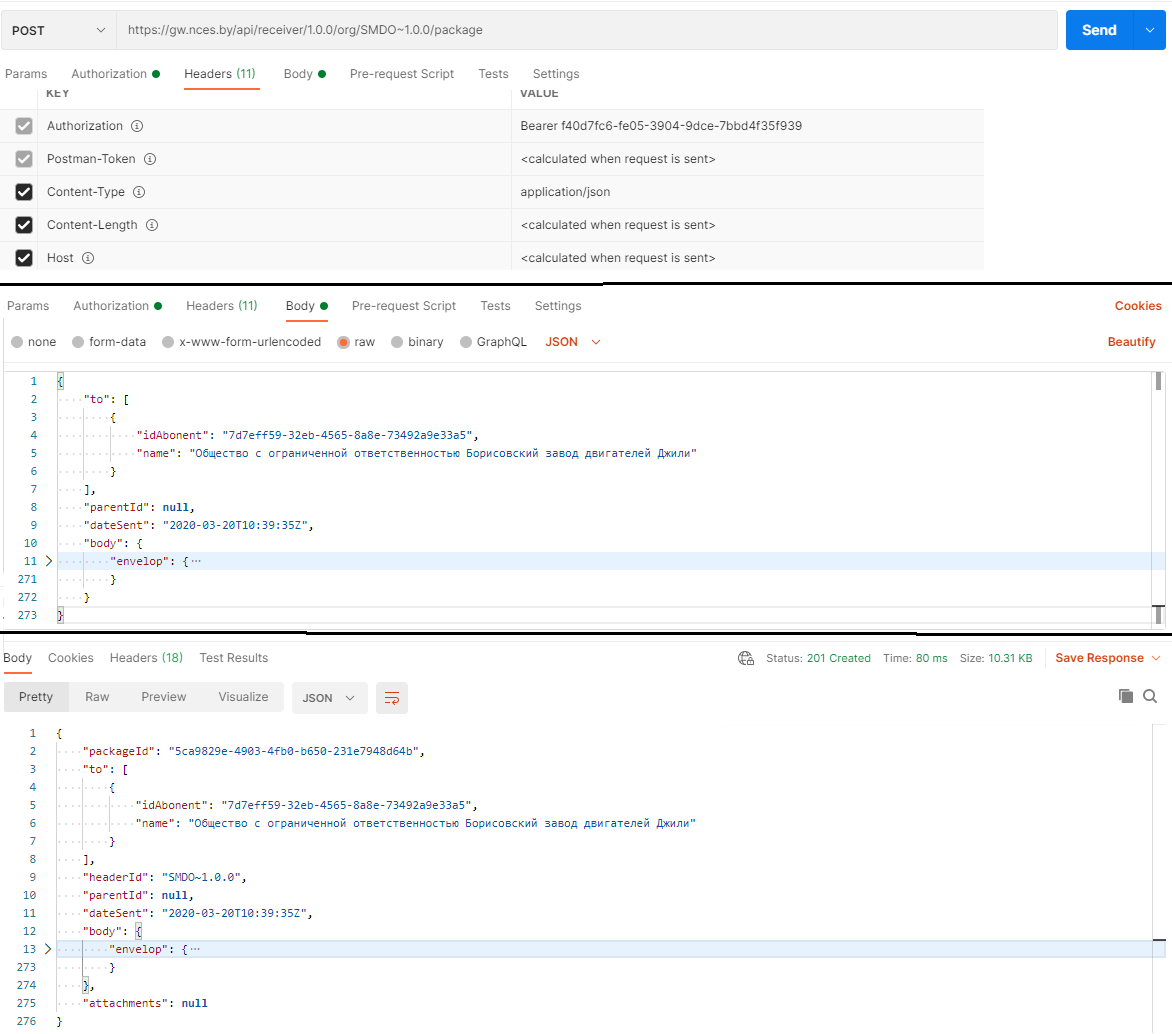
"envelop": {

...

}

}

}'



## 3.1.2 Отправка пакета в маршрутизатор на обработку

После того как был сформирован пакет и загружен в Маршрутизатор, его необходимо отправить на обработку, для этого используется packageId полученный от маршрутизатора при вызове API создания пакета. До момента пока пакет не отправлен на обработку, его можно сравнить с заготовкой или черновиком. Пакет «черновик» не может быть получен абонентами получателями. Пока пакет является «черновиком» его можно модифицировать или удалить, после отправки пакета на обработку он теряет возможность быть отредактированным. Под модификацией пакета понимается: изменение информации, содержащейся в пакете; загрузка/удаление вложений из пакета; изменение подписи у файлов, загруженных в пакет.

Для того, чтобы отправить пакет на обработку в маршрутизатор необходимо вызвать API **GET**[**​/{userType}​/{headerId}​/package​/{packageId}​/send**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/send)используя packageId полученный от маршрутизатора при вызове API создания пакета. Максимальный размер отправляемого пакета не должен превышать 2 Мб. Превышение размера пакета вызовет ошибку (HTTP-код 413 Payload Too Large).

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

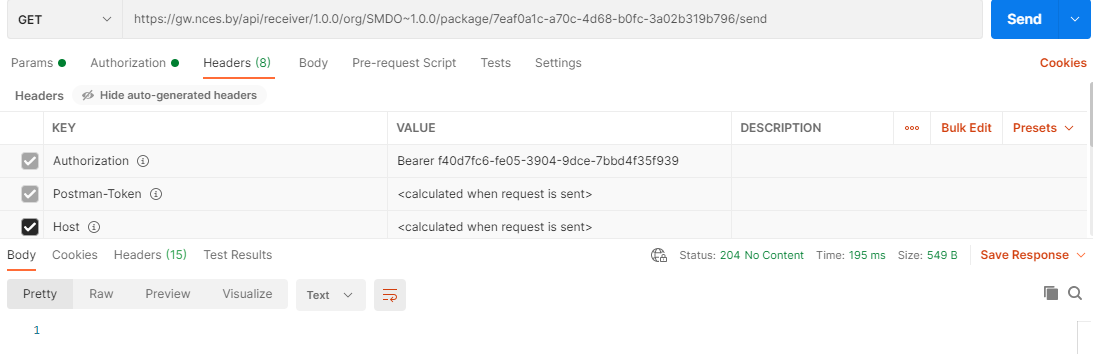
Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 401 | Неавторизованный запрос |
| 403 | Нет доступа |
| 413 | Размер пакета превышает допустимый |
| 500 | Ошибка сервера |

После выполнения запроса отправки ранее созданного пакета на обработку в маршрутизатор:

curl--location--request GET 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/7eaf0a1c-a70c-4d68-b0fc-3a02b319b796/send'\

--header 'Authorization: Bearer f40d7fc6-fe05-3904-9dce-7bbd4f35f939'



## 3.1.3 Получить еще не отправленный пакет

Для того что бы получить еще не отправленный пакет на обработку в маршрутизатор необходимо обратиться по следующему API используя packageId полученный от маршрутизатора при вызове API создания пакета:

**GET**[**​/{userType}​/{headerId}​/package​/{id}**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/get)

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

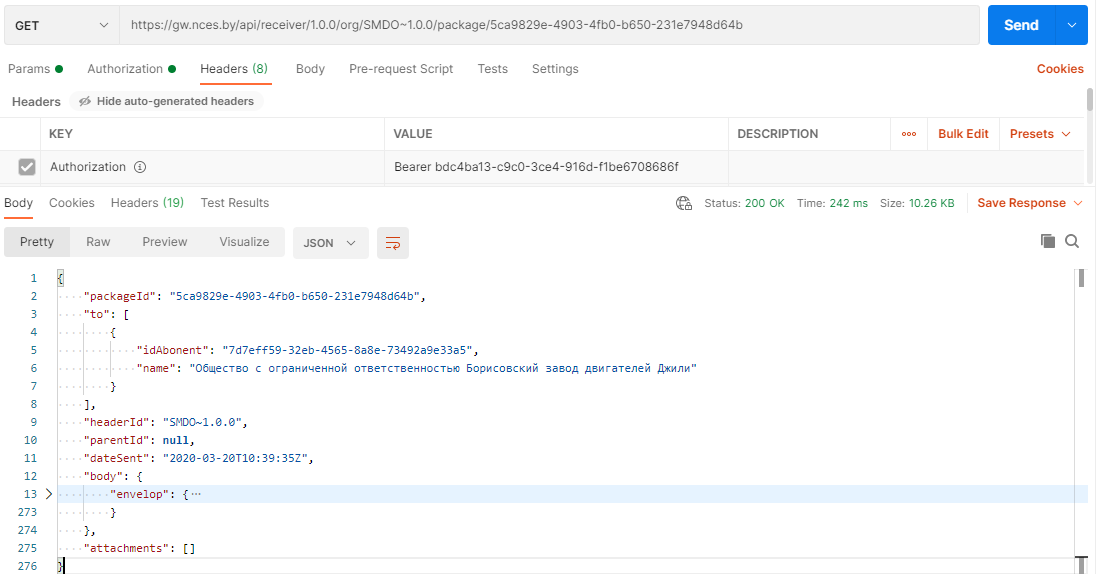
После выполнения запроса на сервере можно ожидать следующий шаблон ответа:

|  |
| --- |
| 200 Успешная операция  {  "packageId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "to": [  {  "idAbonent": "string",  "name": "string"  }  ],  "headerId": "string",  "parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "dateSent": "2021-01-23T18:45:50.890Z",  "body": {...},  "attachments": [  {  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "size": 0,  "name": "string",  "sign": [  {  "keyId": "string",  "signer": "string",  "signtime": "2021-01-23T18:45:50.891Z",  "operationType": "string",  "origSignature": "string",  "data": "string"  }]}]}  } |

Пример запроса на получение еще не отправленного пакета на обработку в маршрутизатор:

curl--location--request GET 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b'\

--header 'Authorization: Bearer bdc4ba13-c9c0-3ce4-916d-f1be6708686f'



## 3.1.4 Обновить информацию в пакете

Для того что бы произвести модификацию информации, в еще не отправленном на обработку пакете, необходимо выполнить запрос по следующему API **PUT**[**​/{userType}​/{headerId}​/package**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/update)передав в теле запроса packageId полученный при формировании пакета.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | application/json |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Тело запроса должно быть сформировано в формате **application/json**, наполнение элемента body определяется форматом обмена, шаблон тела запроса выглядит следующим образом:

{

"packageId": "5ca9829e-4903-4fb0-b650-231e7948d64b",

"to": [

{

"idAbonent": "string",

"name": "string"

}

],

"parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",

"dateSent": "2021-01-23T18:49:29.497Z",

"body": {...}

}

После выполнения запроса возможен следующий шаблон ответа от сервера:

|  |
| --- |
| 200 Успешная операция  {  "packageId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "to": [  {  "idAbonent": "string",  "name": "string"  }  ],  "headerId": "string",  "parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "dateSent": "2021-01-23T18:51:39.411Z",  "body": {...},  "attachments": [  {  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  "size": 0,  "name": "string",  "sign": [  {  "keyId": "string",  "signer": "string",  "signtime": "2021-01-23T18:51:39.411Z",  "operationType": "string",  "origSignature": "string",  "data": "string"  }  ]  }  ]  } |

Пример запроса на обновления информации в еще не отправленном на обработку в маршрутизатор пакете:

curl--location--request PUT 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package'\

--header 'Authorization: Bearer bdc4ba13-c9c0-3ce4-916d-f1be6708686f'\

--header 'Content-Type: application/json'\

--data - raw '{

"packageId": "5ca9829e-4903-4fb0-b650-231e7948d64b",

"to": [

{

"idAbonent": "7d7eff59-32eb-4565-8a8e-73492a9e33a5",

"name": "Общество с ограниченной ответственностью Борисовский завод двигателей Джили"

}

],

"parentId": null,

"dateSent": "2020-03-20T10:39:35Z",

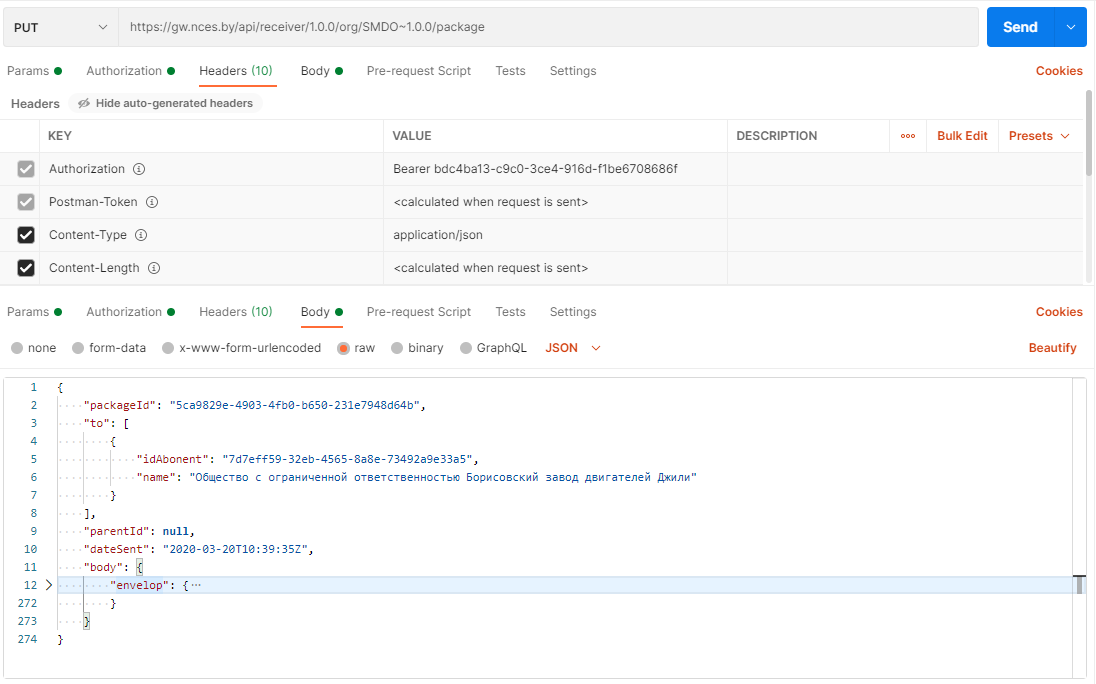
"body": {

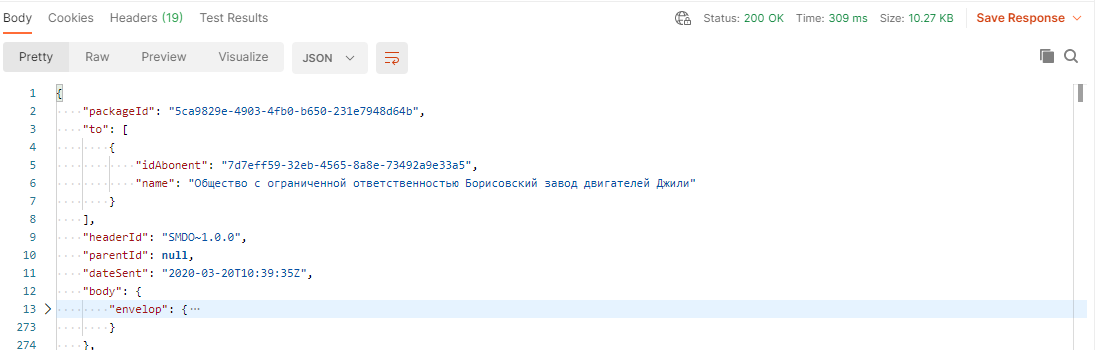
"envelop": {...}

}

}

'





## 3.1.5 Добавить файл к пакету

Для того что бы добавить файл к еще не отправленному на обработку в Маршрутизатор пакету необходимо воспользоваться следующим API

**POST**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/uploadAttach)используя packageId полученный от маршрутизатора при вызове API создания пакета.

Запрос должен быть сформирован в формате multipart/form-data, **за один запрос отсылается только один файл**! Если в одном запросе будет несколько файлов, пользователь получит ошибку в ответе с кодом 400.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | multipart/form-data |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

После выполнения запроса возможен следующий шаблон ответы от сервера:

200 При успешной отправке файла

{

"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",

"size": 0,

"name": "string",

"sign": [

{

"keyId": "string",

"signer": "string",

"signtime": "2021-01-23T18:58:58.714Z",

"operationType": "string",

"origSignature": "string",

"data": "string"

}

]

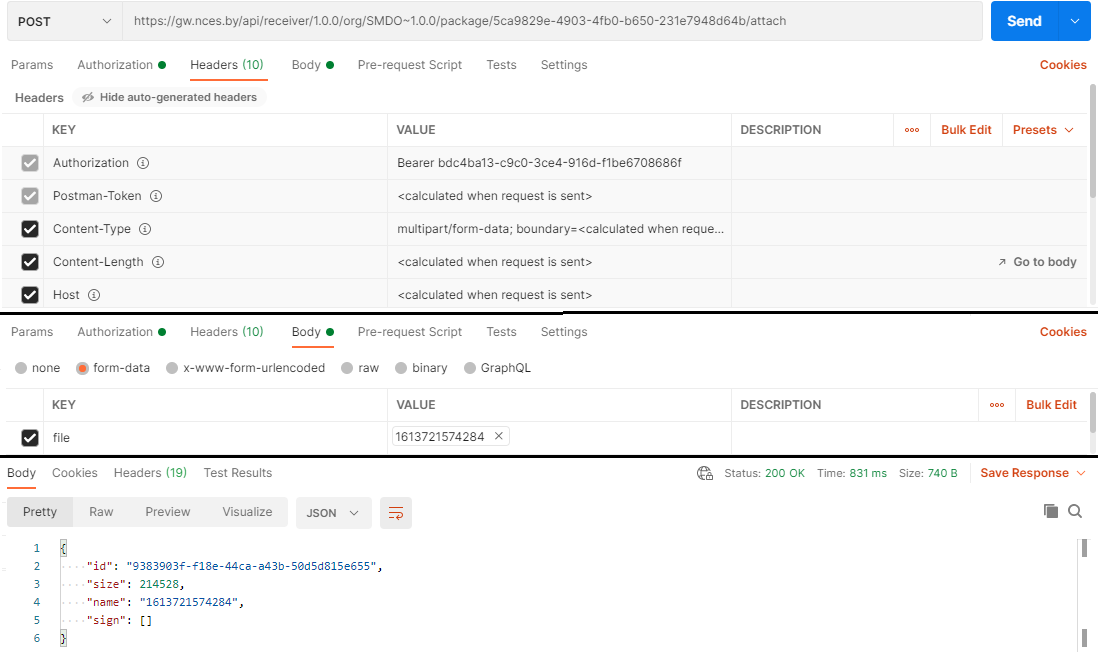
}

Пример добавления вложения в еще не отправленный пакет на обработку:

curl--location--request POST 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b/attach'\

--header 'Authorization: Bearer bdc4ba13-c9c0-3ce4-916d-f1be6708686f'\

--form 'file=@"/C:/Users/xdisa/Desktop/1613721574284"'



## 3.1.6 Удалить файл из пакета

Для того чтобы удалить файл из еще не отправленного пакета на обработку в маршрутизатор, необходимо обратиться к следующему API

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deleteAttachment)

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

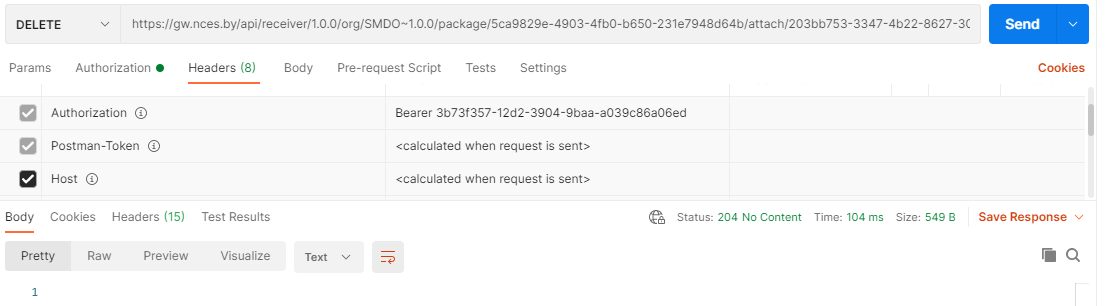
Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Пример запроса на удаление вложения в еще не отправленном пакете:

curl--location—request DELETE 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b/attach/203bb753-3347-4b22-8627-30bd20b2b3c0'\

--header 'Authorization: Bearer 3b73f357-12d2-3904-9baa-a039c86a06ed'



## 3.1.7 Добавить или изменить ЭЦП подписи у файла в пакете

Для того чтобы добавить или изменить ЭЦП подписи у еще не отправленного пакета на обработку в маршрутизатор, необходимо обратиться к следующему API

**PUT**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}​/signs**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/updateAttachmentSigns)

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | application/json |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Для добавления или изменения ЭЦП подписей у файла необходимо сформировать массив и отправить его в формате application/json. Пример шаблона тела запроса представлен ниже:

**[**

**{**

**"keyId": "string",**

**"signer": "string",**

**"signtime": "2021-01-23T19:20:24.950Z",**

**"operationType": "string",**

**"origSignature": "string",**

**"data": "string"**

**}**

**]**

После обращения к API возможен следующий шаблон ответа от сервера:

|  |
| --- |
| 200 Успешная операция  **{**  **"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**  **"size": 0,**  **"name": "string",**  **"sign": [**  **{**  **"keyId": "string",**  **"signer": "string",**  **"signtime": "2021-01-23T19:20:24.950Z",**  **"operationType": "string",**  **"origSignature": "string",**  **"data": "string"**  **}**  **]**  **}** |

Пример запроса на добавление ЭЦП к вложению в пакете:

curl --location --request PUT 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b/attach/9383903f-f18e-44ca-a43b-50d5d815e655/signs' \

--header 'Authorization: Bearer 3b73f357-12d2-3904-9baa-a039c86a06ed' \

--header 'Content-Type: application/json' \

--data-raw '[

{

"keyId": null,

"signer": "TEST",

"signtime": "2020-03-20T10:39:35Z",

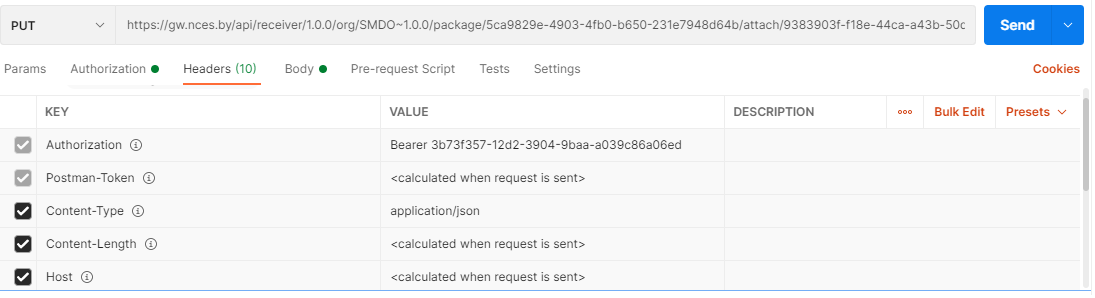
"operationType": null,

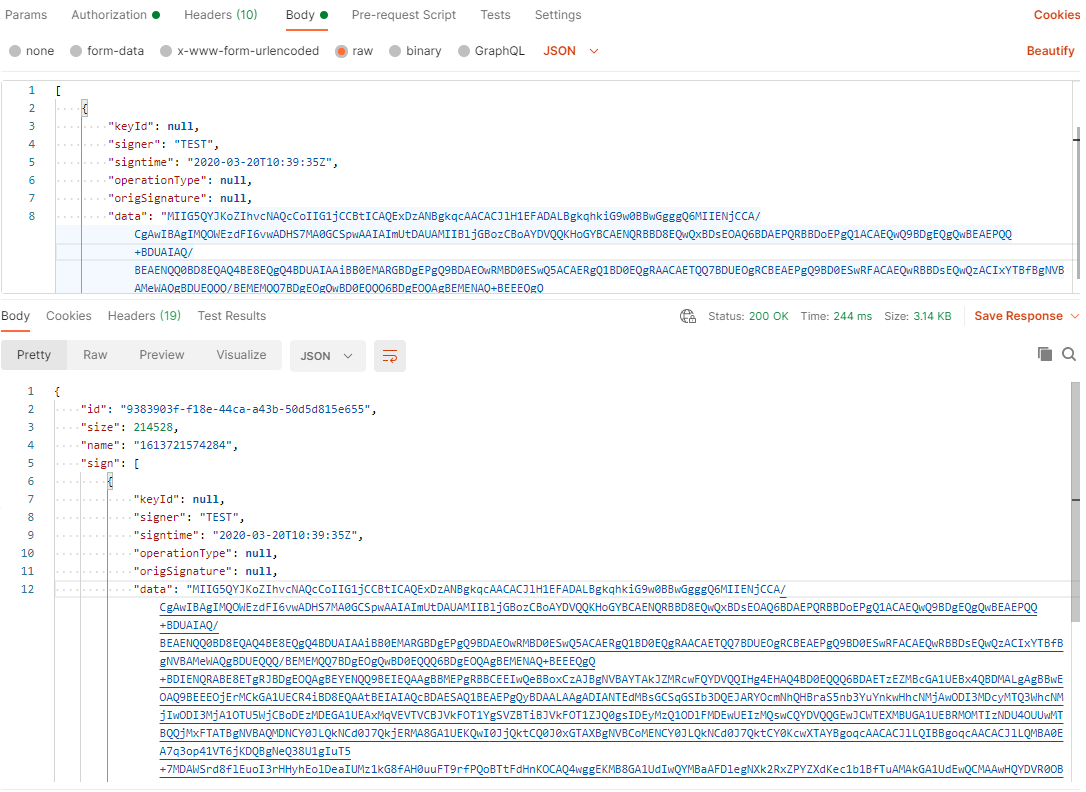
"origSignature": null,

"data": ""

}

]'





## 3.1.8 Удалить ЭЦП у файла

Для того чтобы удалить ЭЦП подписи у файла, в еще не отправленном пакете на обработку в маршрутизатор, необходимо обратиться к API

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}​/attach​/{attachId}​/signs**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deleteAttach)

Параметры запроса берутся из загружаемого пакета в маршрутизатор. Так же их можно узнать воспользовавшись API для получения пакета из раздела 3.1.3.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

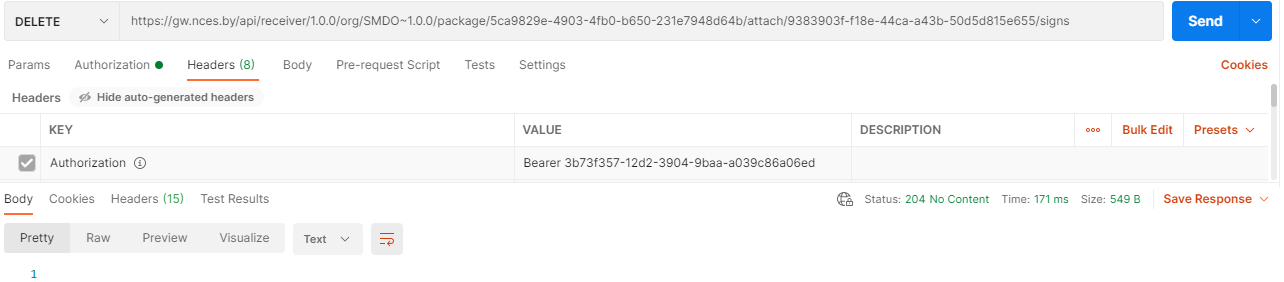
Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Пример запроса на удаление ЭЦП у вложения в еще не отправленном пакете:

curl --location --request DELETE 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b/attach/9383903f-f18e-44ca-a43b-50d5d815e655/signs' \

--header 'Authorization: Bearer 3b73f357-12d2-3904-9baa-a039c86a06ed'



## 3.1.9 Удалить не отправленный на обработку пакет

Для того что бы удалить «черновик» пакета, который еще не был отправлен на дальнейшую обработку в маршрутизатор необходимо обратиться к следующему API

**DELETE**[**​/{userType}​/{headerId}​/package​/{packageId}**](http://172.16.9.47:31106/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B7%D0%BA%D0%B8%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/deletePackageById)

Пакеты, отправленные на обработку, уже не могут быть удалены!

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

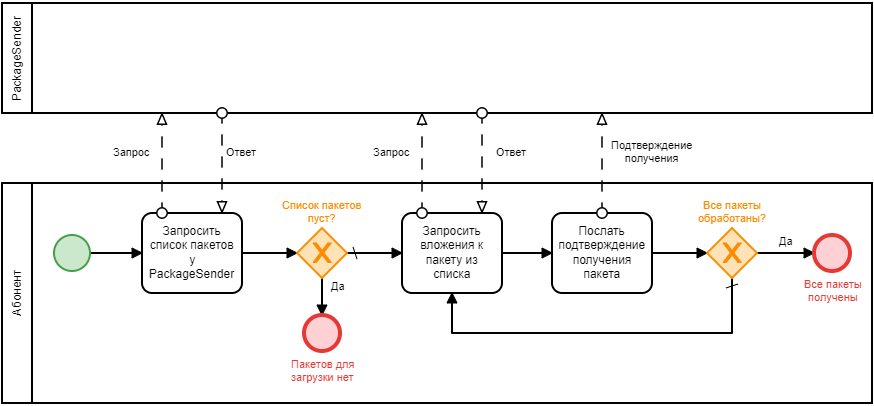
Пример запроса на удаление еще не отправленного в обработку пакета:

curl--location--request DELETE 'https://gw.nces.by/api/receiver/1.0.0/org/SMDO~1.0.0/package/5ca9829e-4903-4fb0-b650-231e7948d64b'\

--header 'Authorization: Bearer 3b73f357-12d2-3904-9baa-a039c86a06ed'



# PackageSender - Процесс получения пакета из Маршрутизатора



Процесс получения пакетов абонентами потребителями начинаться с обращения на соответствующий API сервиса PackageSender. В полученном списке пакетов не содержится вложений. Все вложения в списке представлены метаинформацией. Для получения вложений, относящихся к конкретному пакету, необходимо обратиться на соответствующий API. Для того что бы потребленный пакет не появлялся в списке каждый раз при обращении к сервису PackageSender, необходимо послать подтверждение получения пакета. Более подробная информация о получении пакетов описана в соответствующих разделах данного документа.

## 4.1 Описание модели данных и API сервиса PackageSender

Описание атрибутов модели данных для сервиса PackageSender представлено в таблице 4.1.

Таблица 4.1 - описание модели данных сервиса PackageSender

|  |  |  |
| --- | --- | --- |
| Реквизит | Описание | Формат |
| parentId | Идентификатор родительского пакета | UUID rfc4122 |
| confirmId | Идентификатор потверждения обработки пакета | UUID rfc4122 |
| ack | Признак того является ли пакет квитанцией | boolean |
| dateSent | Дата формирования пакета | ISO 8601 |
| dateIn | Дата поступления пакета в систему | ISO 8601 |
| additionalProperties | Дополнительные параметры | Произвольный JSON объект |
| body | Тело пакета | Произвольный JSON объект |
| from/idAbonent | Идентификатор абонента отправителя | Строка |
| from/name | Имя абонента отправителя | Строка |
| id | Идентификатор пакета | UUID rfc4122 |
| headerId | Заголовок схемы | Строка |
| attachments/id | Идентификатор файла | UUID rfc4122 |
| attachments/size | Размер файла в байтах | Целое число |
| attachments/name | Имя файла | Строка |
| attachments/sign/keyId | Идентификатор открытого ключа подписавшего | Строка |
| attachments/sign/ signer | Имя подписавшего | Строка |
| attachments/sign/ signtime | Дата/время подписи | ISO 8601 |
| attachments/sign/ operationType | Тип операции подписания | Строка |
| attachments/sign/ origSignature | Значение ЭЦП в исходной системе (если выполнялось заверение третьей стороной) | Строка |
| attachments/sign/data | ЭЦП подпись в base64 | Строка |

API сервиса PackageSender позволяет получать пакеты, которые соответствуют указанным критериям фильтрации. Описание параметров фильтра приведено в таблице 4.2.

Таблица 4.2 – описание параметров фильтра

|  |  |  |  |
| --- | --- | --- | --- |
| Реквизит | Описание | Обязательный | Формат |
| page | Номер страницы (значение по умолчанию 0) | - | Целое положительное число |
| size | Размер страницы (значение по умолчанию 20 максимум 100) | - | Целое положительно число |
| onlyNew | Фильтр по только не подтверждённым пакетам (по умолчанию true) | - | boolean |
| dateExec | Фильтр по дате поступление пакетов в систему (если передан, то в ответе будут только пакете с датой больше чем этот параметр, по умолчанию пусто) | - | Iso 8601 |

Реализация списка пакетов представляет собой постраничную модель. На каждой странице может быть определенное количество элементов, за это отвечает параметр «size» в теле запроса. Узнать номер страницы можно из параметра «page». Так как на первой и последней странице количество элементов может быть меньше, чем указано в параметре «size», узнать реальное количество элементов на данной странице можно из параметра «**numberOfElements**». Список не содержит в себе информации о количестве страниц, так как предполагает постраничную обработку, узнать есть ли еще страницы после текущей можно из параметра «**hasNext**».

## 4.2.1 Получение списка пакетов

Для получения списка пакетов необходимо обратиться к Rest API

**POST**[**​/{userType}​/{headerId}​/package**](http://172.16.9.47:31117/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/getPage)передав при этом необходимые фильтры в теле запроса с типом application/json.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | application/json |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Пример шаблона тела запроса:

**{**

**"page": 0,**

**"size": 50,**

**"onlyNew": true,**

**"dateExec": "2021-01-23T20:31:50.217Z"**

**}**

После обработки запроса возможен следующий шаблон ответа:

|  |
| --- |
| 200 Успешная операция  **{**  **"number": 0,**  **"size": 50,**  **"numberOfElements": 0,**  **"hasNext": true,**  **"content": [**  **{**  **"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**  **"confirmId": "42fsd454-5717-4862-b3fc-233asdfa66afa6",**  **"from": {**  **"idAbonent": "string",**  **"typeAbonent": "string",**  **"name": "string",**  **"orgId": "string",**  **"applicationId": "string",**  **"tags": [**  **0**  **]**  **},**  **"to": {**  **"idAbonent": "string",**  **"typeAbonent": "string",**  **"name": "string",**  **"orgId": "string",**  **"applicationId": "string",**  **"tags": [**  **0**  **]**  **},**  **"headerId": "string",**  **"ack": false,**  **"parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**  **"attachments": [**  **{**  **"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**  **"size": 0,**  **"name": "string",**  **"antivirus": true,**  **"antivirusStmp": "2021-01-23T20:31:50.246Z",**  **"eds": true,**  **"edsStmp": "2021-01-23T20:31:50.247Z",**  **"sign": [**  **{**  **"keyId": "string",**  **"signer": "string",**  **"signtime": "2021-01-23T20:31:50.247Z",**  **"operationType": "string",**  **"origSignature": "string",**  **"data": "string"**  **}**  **]**  **}**  **],**  **"dateIn": "2021-01-23T20:31:50.247Z",**  **"dateSent": "2021-01-23T20:31:50.247Z",**  **"body": {**  **"additionalProp1": {},**  **"additionalProp2": {},**  **"additionalProp3": {}**  **}**  **}**  **]**  **}** |
| Пример получения списка пакетов: |

curl --location --request POST 'https://gw.nces.by/api/sender/1.0.0/org/SMDO~1.0.0/package' \

--header 'Authorization: Bearer ca970deb-b878-313b-9b4e-62895a6f625c' \

--header 'Content-Type: application/json' \

--data-raw '{

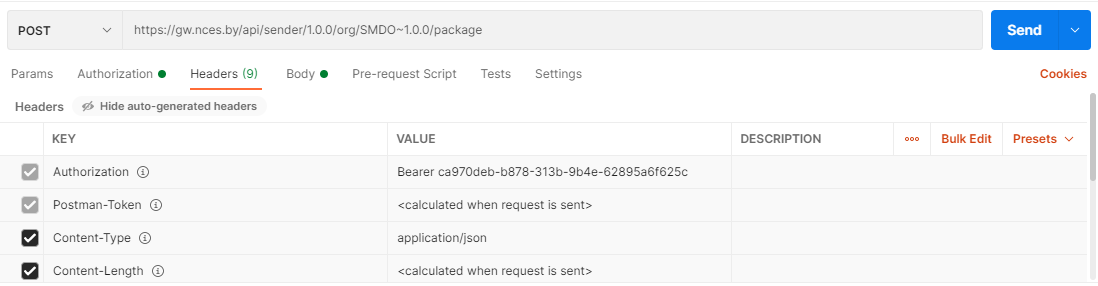
    "page": 0,

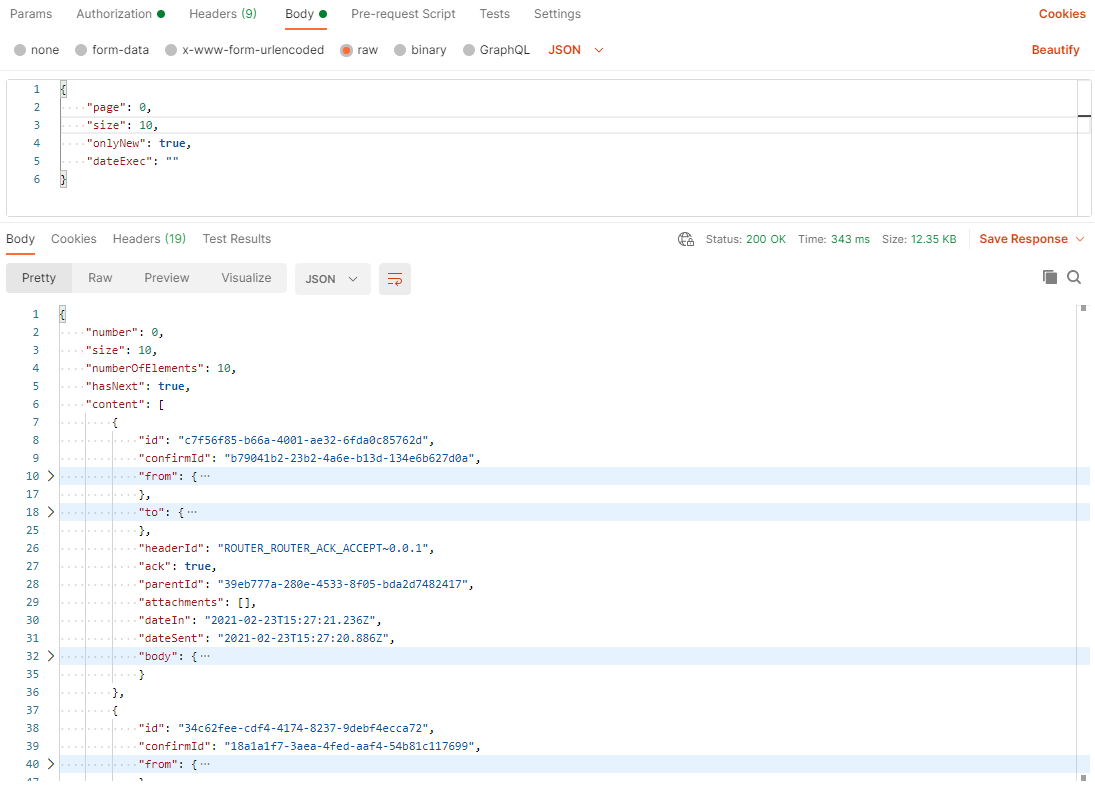
    "size": 10,

    "onlyNew": true,

    "dateExec": ""

}'





## 4.2.2 Получение вложения относящегося к пакету

Получение файла происходит через поток **application/octet-stream**.

Для получения вложения относящегося к пакету необходимо послать запрос на API **GET**[**​/{userType}​/{headerId}​/package​/{id}​/attach​/{attachId}**](http://172.16.9.47:31117/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/downloadFile)

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 200 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

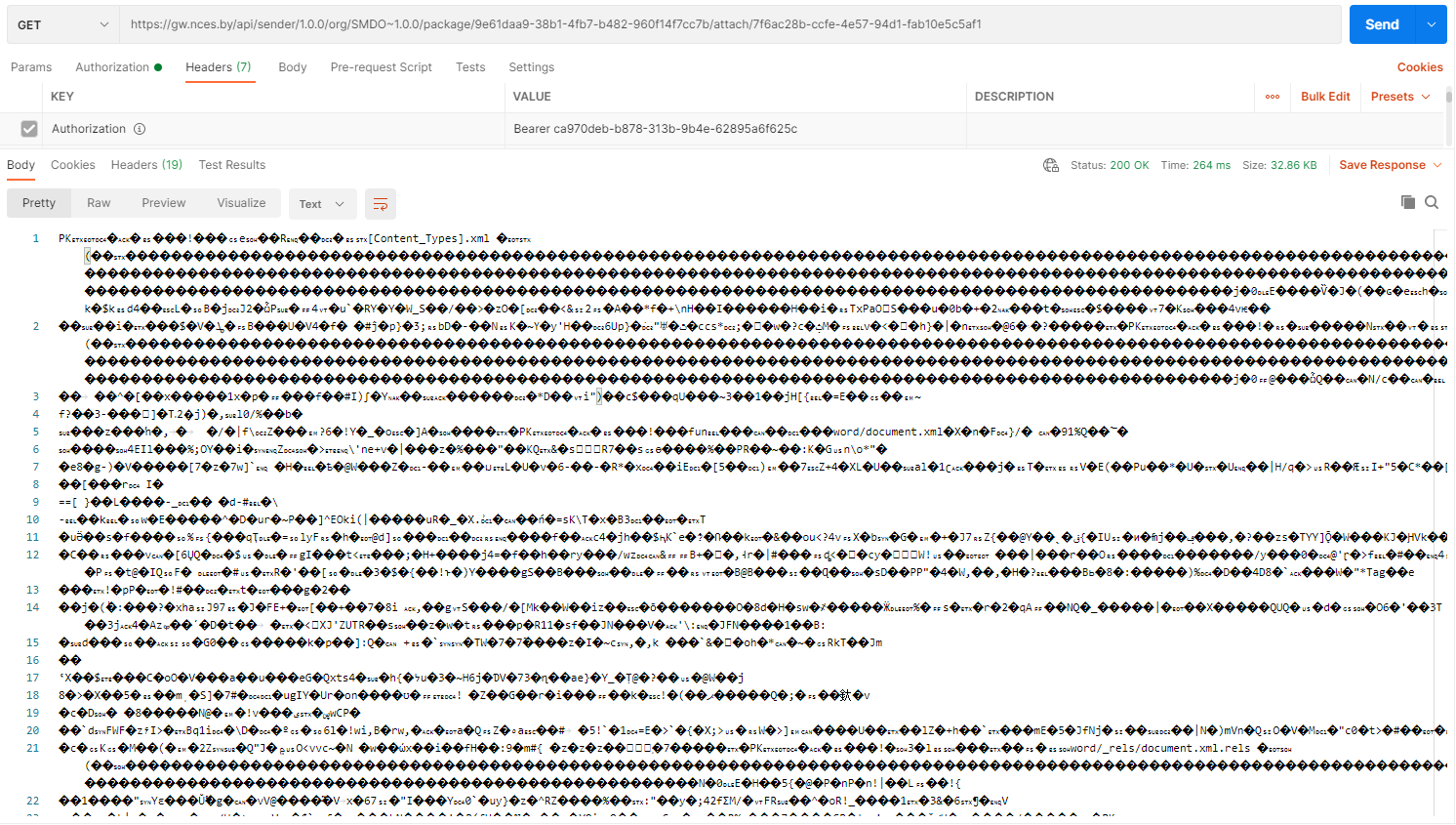
После положительной обработки запроса на сервере будет получен ответ в виде Content-type application/octet-stream:

|  |
| --- |
| 200 Успешная операция  **String($binary)** |

Пример запроса получения вложения из пакета:

curl--location--request GET 'https://gw.nces.by/api/sender/1.0.0/org/SMDO~1.0.0/package/9e61daa9-38b1-4fb7-b482-960f14f7cc7b/attach/7f6ac28b-ccfe-4e57-94d1-fab10e5c5af1'\

--header 'Authorization: Bearer ca970deb-b878-313b-9b4e-62895a6f625c'



## 4.2.3 Подтверждение получения списка пакетов

Если есть необходимость подтвердить получение сразу нескольких пакетов, то необходимо обратиться на API сервиса PackageSender

**POST**[**​/{userType}​/{headerId}​/package​/confirm​/batch**](http://172.16.9.47:31117/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/confirmBatch)и передать массив идентификаторов подтверждений получений пакетов в типе **application/json**.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |
| Content-Type | application/json |
| Content-Length | Размер тела запроса в байтах |

Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Для подтверждения получения списка пакетов необходимо передать массив значений идентификаторов из ключей **confirmId**, которые можно узнать при получении списка пакетов.

Пример шаблона тела запроса представлен ниже:

**{**

**"ids": [**

**"3fa85f64-5717-4562-b3fc-2c963f66afa6"**

**]**

**}**

Пример запроса подтверждения получения списка пакетов:

curl--location--request POST 'https://gw.nces.by/api/sender/1.0.0/org/SMDO~1.0.0/package/confirm/batch'\

--header 'Authorization: Bearer ca970deb-b878-313b-9b4e-62895a6f625c'\

--header 'Content-Type: application/json'\

--data - raw '{

"ids": [

"1a402dfa-38df-401c-80f7-d402dda9de99",

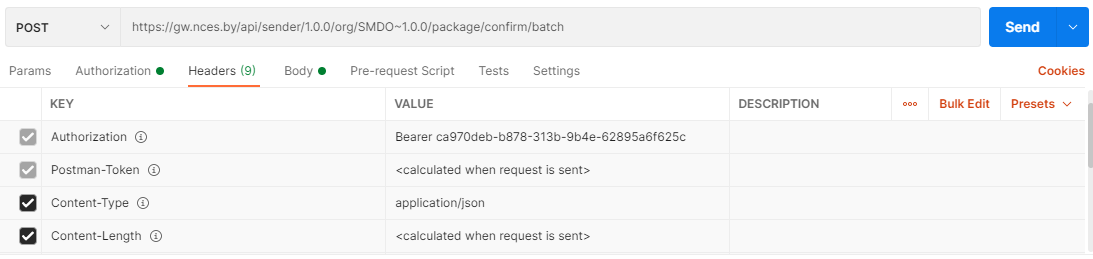
"fc4a0aaf-cb80-4ba3-a7c2-b429cc0a846d",

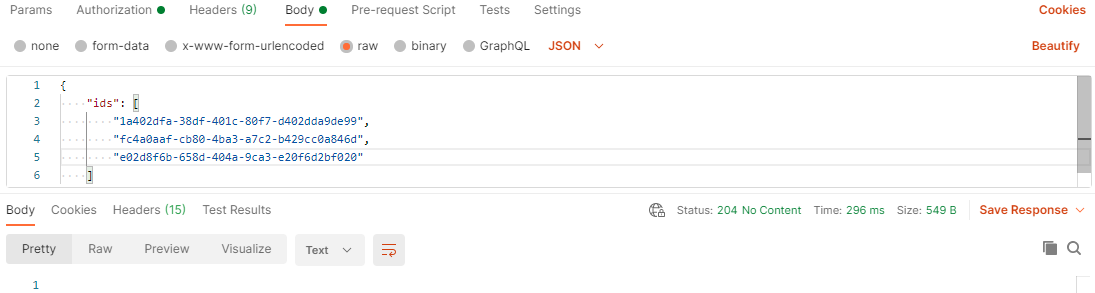
"e02d8f6b-658d-404a-9ca3-e20f6d2bf020"

]

}

'





## 4.2.4 Подтверждение получения одного пакета

Для того что бы подтвердить получение конкретного пакета необходимо отправить запрос на API сервиса PackageSender

**GET**[**​/{userType}​/{headerId}​/package​/confirm​/{confirmId}**](http://172.16.9.47:31117/swagger-ui/index.html?url=/v3/api-docs&validatorUrl=#/API%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D1%8F%20%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%BE%D0%B2%20%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F%D0%BC%D0%B8/confirm)

Параметр запроса **confrimId** можно узнать при получении списка пакетов.

Заголовки запроса

|  |  |
| --- | --- |
| Наименование | Описание |
| Authorization | Bearer token |

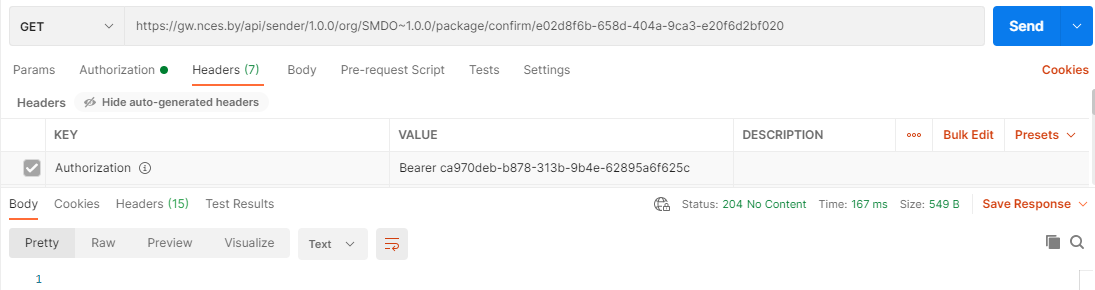
Коды ответов

|  |  |
| --- | --- |
| Наименование | Описание |
| 204 | Успешная операция |
| 400 | Неверный запрос |
| 403 | Нет доступа |
| 404 | Не найдено |
| 500 | Ошибка сервера |

Пример запроса подтверждения получения одного пакета:

curl--location--request GET 'https://gw.nces.by/api/sender/1.0.0/org/SMDO~1.0.0/package/confirm/e02d8f6b-658d-404a-9ca3-e20f6d2bf020'\

--header 'Authorization: Bearer ca970deb-b878-313b-9b4e-62895a6f625c'



## 4.3 Квитанции

Квитанция представляет собой обычный пакет маршрутизатора, который служит для уведомлений о каких-то событиях. Все квитанции делятся на типы:

- о принятии в ядро;

- о принятии абонентом;

- об ошибке;

- пользовательская квитанция;

Зависит от конкретной схемы форматов обмена какие виды квитанций можно получить. Общая структура квитанций одинакова, а вот тело (body -полезная информация) может отличаться. Определить является ли пакет квитанцией можно по параметру «ack» имеющий значение «true».

Пример шаблона квитанции:

**{**

**"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**

**"confirmId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**

**"from": {**

**"idAbonent": "string",**

**"typeAbonent": "string",**

**"name": "string",**

**"orgId": "string",**

**"applicationId": "string",**

**"tags": [**

**0**

**]**

**},**

**"to": {**

**"idAbonent": "string",**

**"typeAbonent": "string",**

**"name": "string",**

**"orgId": "string",**

**"applicationId": "string",**

**"tags": [**

**0**

**]**

**},**

**"headerId": "string",**

**"ack": true,**

**"parentId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**

**"attachments": [**

**{**

**"id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",**

**"size": 0,**

**"name": "string",**

**"antivirus": true,**

**"antivirusStmp": "2021-01-23T17:42:56.274Z",**

**"eds": true,**

**"edsStmp": "2021-01-23T17:42:56.274Z",**

**"sign": [**

**{**

**"keyId": "string",**

**"signer": "string",**

**"signtime": "2021-01-23T17:42:56.274Z",**

**"operationType": "string",**

**"origSignature": "string",**

**"data": "string"**

**}**

**]**

**}**

**],**

**"dateIn": "2021-01-23T17:42:56.274Z",**

**"dateSent": "2021-01-23T17:42:56.274Z",**

**"body": {**

**"additionalProp1": {},**

**"additionalProp2": {},**

**"additionalProp3": {}**

**}**

**}**

# Взаимодействие с СМДО

СМДО - система межведомственного электронного документооборота государственных органов Республики Беларусь, обеспечивающая информационное взаимодействие систем электронного документооборота.

Для осуществления обмена информацией используется единый формат обмена, к которому приводятся сообщения принимаемые и отправляемые СЭД.

В будущем абоненты СМДО должны будут перейти на использование Маршрутизатора АИСМВ. В период, когда абоненты будут переходить от использования СМДО к Маршрутизатору АИСМВ данные системы могут существовать параллельно.

Абоненты, подключенные к СМДО, обмениваются между собой пакетами в формате **XML**. В то же время, Абоненты подключенные к «АПК Платформа» и использующие Маршрутизатор АИСМВ обмениваются пакетами в формате **JSON**. В виду разных подходов и среды в целом, для обеспечения обмена сообщениями в переходный период между абонентами СМДО и АИСМВ разработан сервис конвертор пакетов из XML в JSON и в обратную сторону из JSON в XML. Конвертация пакетов происходит в автоматическом режиме в рамках взаимодействия система-система (СМДО-Маршрутизатор).