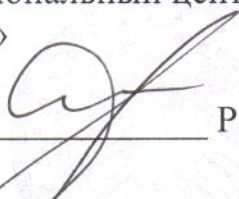


УТВЕРЖДАЮ

Директор республиканского
унитарного предприятия
«Национальный центр электронных
услуг»



Р.А. Градусов
«__» _____ 2021 г.

СОГЛАСОВАНО

Председатель Государственного
таможенного комитета
Республики Беларусь



В.Н. Орловский
«__» _____ 2021 г.

Технические условия

**для подключения информационных систем заинтересованных лиц к
информационной системе таможенных органов через
общегосударственную автоматизированную информационную систему
(при передаче заявления о совершении операций в отношении
находящихся за пределами таможенной территории Евразийского
экономического союза транспортных средств международной перевозки,
являющихся товарами, помещенными под таможенную процедуру
временного ввоза (допуска))**

Версия 1.0

МИНСК
2021

СОДЕРЖАНИЕ

1. Сокращения.....	3
2. Описание веб-сервиса.....	3
3. Описание операций веб-сервисов.....	4
3.1. Отправка документа в ОАИС	4
3.2. Получение списка запросов пользователя	6
3.2.1. По диапазону	6
3.2.2. По дате обновления.....	8
3.2.3. По диапазону дат обновления.....	8
3.2.4. По регистрационному номеру	9
3.2.5. По GUID файла.....	9
3.3. Получение информации о запросе пользователя	9
3.4. Получение списка сообщений, связанных с запросом.....	11
3.5. Получение содержимого сообщения	11
4. Описание кодов состояния НТТР	12
5. Описание порядка взаимодействия.....	13
5.1. Сценарий информационного обмена при передаче ЭТД	13
5.1.1. Формирование и подписание документа в ИСЗЛ.....	14
5.1.2. Подача документа в ОАИС	14
5.1.3. Передача и обработка документа в ИСТО. Передача сообщений, сформированных ИСТО, в ОАИС	15
5.1.4. Получение ИСЗЛ таможенных сообщений из ОАИС.....	16
6. Правила использования ЭЦП при подписании ЭТД.....	19
6.1. Общие положения	19
6.2. Порядок формирования ЭЦП	19
6.3. Порядок проверки ЭЦП.....	20
6.4. Использование ЭЦП в прикладных электронных документах	21
6.5. Нормативные документы по ЭЦП.....	22
7. Требования при взаимодействии с ОАИС	22
Приложение А	23
Приложение Б.....	28

Аннотация

Настоящие технические условия разработаны с целью обеспечения взаимодействия между информационными системами заинтересованных лиц и информационной системой таможенных органов посредством общегосударственной автоматизированной информационной системы и определяют порядок передачи информации, форматы и структуры передаваемых данных, а также смысловое содержание кодов информации.

Технические условия могут быть расширены в процессе внедрения и развития системы.

1. Сокращения

Используемые сокращения приведены в таблице 1.1.

Таблица 1.1

Сокращение	Расшифровка
API	Application Programming Interface
GUID	Globally Unique Identifier - статистически уникальный 128-битный идентификатор
HTTP	HyperText Transfer Protocol – протокол передачи гипертекста
REST	Representational State Transfer – архитектурный стиль взаимодействия компонентов распределённого приложения в сети
XML	eXtensible Markup Language - расширяемый язык разметки
ЭЦП	Электронная цифровая подпись
ОАИС	Общегосударственная автоматизированная информационная система
ЭТД	Электронный таможенный документ
ИСЗЛ	Информационная система заинтересованных лиц
ИСТО	Информационная система таможенных органов
ЗСО	Заявление о совершении операций в отношении временно вывезенных транспортных средств международной перевозки

2. Описание веб-сервиса

Используется архитектурный стиль REST. В рамках REST взаимодействий транзакции происходят на сервере с помощью HTTP запросов и ответов.

Абсолютная часть указателя ресурса имеет вид:

https://<host>:<port>/ServiceISZL/ecd/v1 ,

где:

- https – сетевой протокол, используемый для обращения к ресурсу;
- host – полностью прописанное доменное имя хоста в системе DNS или IP-адрес хоста в формате четырех групп десятичных чисел, разделенных точками; числа – целые в интервале от 0 до 255;
- port – порт хоста для подключения;
- ServiceISZL – идентификатор приложения;
- ecd – наименование ресурса;
- v1 – постоянная часть указателя, определяющая версию API.

Абсолютная часть остается неизменной для ресурсов веб-сервиса. Используемая кодировка для запросов – UTF-8. В ответных сообщениях от API в заголовке Content-Language будет указан используемый язык (например, ru).

При вызове методов сервиса в заголовке запроса необходимо передавать:

- токен авторизации ИСЗЛ (выдается после успешного прохождения тестирования взаимодействия ИСЗЛ с ОАИС);
- идентификационный номер инициатора услуги (выдается каждому пользователю при заключении договора на оказание услуги).

Список передаваемых в заголовке запроса параметров приведен в таблице 2.1.

Таблица 2.1

Название параметра заголовка	Описание параметра	Значение	Обязательное
Authorization	Токен авторизации ИСЗЛ	Bearer <токен>	Да
UserId	Идентификационный номер инициатора услуги	строка	Да

При передаче неверного токена авторизации будет возвращен HTTP код 401, а тело ответа содержать следующее сообщение:

```
<ams:fault xmlns:ams="http://wso2.org/apimanager/security">
  <ams:code>900901</ams:code>
  <ams:message>Invalid Credentials</ams:message>
  <ams:description>Access failure for API: /ServiceISZL/ecd/v1, version: v1
status: (900901) - Invalid Credentials. Make sure you have given the correct
access token</ams:description>
</ams:fault>
```

3. Описание операций веб-сервисов

3.1. Отправка документа в ОАИС

Относительная часть идентификатора ресурса имеет вид:

POST /request/{file_guid}?pto_id={pto_id}&remark={remark} .

При отправке запроса необходимо указать заголовок «Content-Type: application/xml». В теле запроса необходимо передавать содержимое документа в формате XML:

```
<?xml version="1.0" encoding="utf-8"?>
<TMPA>
  <Declarant ID=" ... ">
    Тело ЭТД в соответствии со структурой XML-документа
  </Declarant>
  <Signature> ... </Signature>
</TMPA>
```

Тэг <Signature> является обязательным в случае использования ЭЦП при подписании документа.

Перед передачей документа, ему необходимо присвоить GUID - уникальный идентификатор документа длиной 36 символов, структура которого приведена на рисунке 1.

[x x x x x x x x]—[x x x x]—[x x x x]—[x x x x]—[x x x x x x x x x x x x]
 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36

Рисунок 1 – Структура уникального идентификатора

Метод осуществляет загрузку ЭТД в ОАИС. При успешном сохранении документа метод возвращает информацию о созданном объекте.

Входные параметры приведены в таблице 3.1.

Таблица 3.1

Имя	Описание	Значение	Обязательное
pto_id	Код таможенного органа	число	Да
file_guid	GUID отправляемого файла	GUID	Да
remark	Исходящий номер документа	строка	Нет

Структура возвращаемого сообщения:

```
{
  "request": [
    {
      "id": "...",
      "status_id": "...",
      "date_update": "...",
      "comment": "..."
    }
  ]
}
```

Описание полей возвращаемого сообщения приведено в таблице 3.2.

Таблица 3.2

Имя	Описание	Значение	Мн.
id	Идентификатор запроса в ОАИС	число	[1]
status_id	Код статуса запроса	число	[1]
date_update	Дата и время обновления записи	дата (YYYY-MM-DDThh:mm:ss)	[1]
comment	Замечания при принятии документа в ОАИС	строка	[0..1]

Список кодов статусов запроса приведен в таблице 3.3.

Таблица 3.3

Код статуса	Описание статуса
0	документ передан в ОАИС и ожидает отправки в ИСТО
1	документ передан в ИСТО и находится в обработке
2	отказ принятия документа в ИСТО
3	успешное принятие документа в ИСТО
5	документ зарегистрирован в ИСТО
9	ошибка обработки
11	отказ в регистрации (возврате) документа

3.2. Получение списка запросов пользователя

3.2.1. По диапазону

Относительная часть идентификатора ресурса имеет вид:

GET /requests?offset={offset}&limit={limit} .

Метод возвращает отсортированный по убыванию даты отправления список запросов данного пользователя с заданным смещением. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.4.

Таблица 3.4

Имя	Описание	Значение	Обязательное
offset	Смещение относительно последней отправленной записи	число, по умолчанию 0	Нет
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения:

```

{
  "requests": [
    {
      "id": "...",
      "status_id": "...",
      "file_guid": "...",
      "ed_type": "...",
      "date_of": "...",
      "date_update": "...",
      "reg_no": "...",
    }
  ]
}

```

```

        "app_no": "...",
        "date_reg": "...",
        "date_app": "...",
    },
]
}

```

Описание полей возвращаемого сообщения приведено в таблице 3.5.

Таблица 3.5

Имя	Описание	Значение	Мн.
id	Идентификатор запроса	число	[1]
status_id	Код статуса запроса	число	[1]
file_guid	GUID отправленного файла	GUID	[1]
ed_type	Тип ЭТД	строка «ЗСО»	[1]
date_of	Дата и время отправления документа в ОАИС	дата (YYYY-MM-DDThh:mm:ss)	[1]
date_update	Дата и время обновления записи	дата (YYYY-MM-DDThh:mm:ss)	[1]
reg_no	Регистрационный номер документа	строка	[0..1]
app_no	Сведения о выпуске /Регистрационный номер документа	строка	[0..1]
date_reg	Дата и время регистрации документа	дата (YYYY-MM-DDThh:mm:ss)	[0..1]
date_app	Дата и время выпуска/Дата и время регистрации документа	дата (YYYY-MM-DDThh:mm:ss)	[0..1]

3.2.2. По дате обновления

Относительная часть идентификатора ресурса имеет вид:

GET /requests?date_update={date_update}&limit={limit} .

Метод возвращает список запросов данного пользователя, которые были обновлены после переданной даты. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.6.

Таблица 3.6

Имя	Описание	Значение	Обязательное
date_update	Дата и время обновления последнего полученного запроса	дата (YYYY-MM-DDThh:mm:ss)	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.2.3. По диапазону дат обновления

Относительная часть идентификатора ресурса имеет вид:

GET /requests?date_from={date_from}&date_to={date_to} .

Метод возвращает список запросов данного пользователя, которые были обновлены в указанный период. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.7.

Таблица 3.7

Имя	Описание	Значение	Обязательное
date_from	Дата и время начала диапазона	дата (YYYY-MM-DDThh:mm:ss)	Да
date_to	Дата и время конца диапазона	дата (YYYY-MM-DDThh:mm:ss)	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.2.4. По регистрационному номеру

Относительная часть идентификатора ресурса имеет вид:

GET /requests?reg_no={reg_no}&limit={limit} .

Метод возвращает список сообщений данного пользователя, содержащие заданный регистрационный номер. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.8.

Таблица 3.8

Имя	Описание	Значение	Обязательное
reg_no	Регистрационный номер документа	строка	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.2.5. По GUID файла

Относительная часть идентификатора ресурса имеет вид:

GET /requests?file_guid={file_guid}&limit={limit} .

Метод возвращает список запросов данного пользователя, содержащие заданный GUID файла. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.9.

Таблица 3.9

Имя	Описание	Значение	Обязательное
file_guid	GUID отправленного файла	GUID	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.3 Получение информации о запросе пользователя

Относительная часть идентификатора ресурса имеет вид:

GET /request/{rq_id} .

Метод возвращает информацию по заданному запросу пользователя.

Входные параметры приведены в таблице 3.10.

Таблица 3.10

Имя	Описание	Значение	Обязательное
rq_id	Идентификатор запроса	число	Да

Структура возвращаемого сообщения:

```

{
  "requests": {
    "id": "...",
    "status_id": "...",
    "file_guid": "...",
    "ed_type": "...",
    "date_of": "...",
    "date_update": "...",
    "reg_no": "...",
    "app_no": "...",
    "date_reg": "...",
    "date_app": "...",
  }
}

```

Описание полей возвращаемого сообщения приведено в таблице 3.11.

Таблица 3.11

Имя	Описание	Значение	Мн.
id	Идентификатор запроса	число	[1]
status_id	Код статуса запроса	число	[1]
file_guid	GUID отправленного файла	GUID	[1]
ed_type	Тип ЭТД	строка «ЗСО»	[1]
date_of	Дата и время отправления документа в ОАИС	дата (YYYY-MM-DDThh:mm:ss)	[1]
date_update	Дата и время обновления записи	дата (YYYY-MM-DDThh:mm:ss)	[1]
reg_no	Регистрационный номер документа	строка	[0..1]
app_no	Сведения о выпуске /Регистрационный номер документа	строка	[0..1]
date_reg	Дата и время регистрации документа	дата (YYYY-MM-DDThh:mm:ss)	[0..1]

date_app	Дата и время выпуска/Дата и время регистрации документа	дата (YYYY-MM-DDThh:mm:ss)	[0..1]
----------	---	----------------------------	--------

3.4. Получение списка сообщений, связанных с запросом

Относительная часть идентификатора ресурса имеет вид:

GET /files/{rq_id} .

Метод возвращает список объектов, содержащих код вида документа, дату создания, тип, идентификатор сообщений, относящихся к заданному запросу пользователя.

Входные параметры приведены в таблице 3.12.

Таблица 3.12

Имя	Описание	Значение	Обязательное
rq_id	Идентификатор запроса	число	Да

Структура возвращаемого сообщения:

```

{
  "files": [
    {
      "ln_id": "...",
      "date_of": "...",
      "ln_type": "..."
    },
  ]
}

```

Описание полей возвращаемого сообщения приведено в таблице 3.13.

Таблица 3.13

Имя	Описание	Значение	Мн.
ln_id	Идентификатор связанного сообщения	число	[1]
date_of	Дата и время создания сообщения	дата (YYYY-MM-DDThh:mm:ss)	[1]
ln_type	Тип сообщения	число (см. таблицу 5.1)	[1]

3.5. Получение содержимого сообщения

Относительная часть идентификатора ресурса имеет вид:

GET /file/{ln_id} .

Метод возвращает содержимое xml-сообщения (декларантов, таможни, технологических), имеющего заданный идентификатор и относящегося к заданному пользователю.

Входные параметры приведены в таблице 3.14.

Таблица 3.14

Имя	Описание	Значение	Обязательное
-----	----------	----------	--------------

ln_id	Идентификатор сообщения	число	Да
-------	-------------------------	-------	----

В теле ответа будет возвращено содержимое запрашиваемого сообщения в формате XML и с Content-Type: application/xml.

4. Описание кодов состояния HTTP

При работе веб-сервисов используются следующие интерпретации кодов состояния HTTP.

200 OK – успешный запрос. Если клиентом были запрошены какие-либо данные, то они находятся в теле ответа.

400 Bad Request – ошибка запроса, неверные параметры запроса либо запрос не может быть выполнен.

401 Unauthorized – не удалось произвести авторизацию.

403 Forbidden – запрос производится к ресурсу, к которому нет доступа.

404 Not Found – запрос производится к ресурсу, который не найден.

429 Too Many Requests – производится слишком много запросов.

502 Bad Gateway – шлюз вернул некорректный ответ.

503 Service Unavailable – сервис недоступен.

504 Gateway Timeout - шлюз не отвечает.

500 Internal Server Error – при выполнении запроса произошла ошибка. Описание данной ошибки будет отправлено в теле ответа и имеет следующую структуру:

```
{
  "errId": "...",
  "errDescr": "..."
}
```

где:

- errId - код результата обработки;
- errDescr - описание результата обработки.

Примеры значений параметров errId и errDescr приведены в таблице 4.1.

Таблица 4.1

Код результата обработки (errId)	Описание результата обработки (errDescr)
2	Неверный код вида документа.
3	Передача документов данного типа запрещена для текущего пользователя.

6	Пользователь заблокирован.
10	Документ с данным идентификатором файла передан в ОАИС ранее. Для повторной отправки документа создайте документ с новым идентификатором файла.
12	Документ не подписан.
22	Выявлено несоответствие лица, представляющего документ, с лицом, указанным в качестве декларанта таможенной процедуры, либо являющимся таможенным представителем. Документ в таможенный орган не направлен. Представьте документ с корректно заполненными сведениями.
23	Выявлено несоответствие лица, представляющего электронный документ, с лицом, указанным в сертификате открытого ключа. Электронный документ в таможенный орган не направлен. Представьте электронный документ с корректно заполненными сведениями.
24	Выявлено несоответствие лица, представляющего электронный документ, с лицом, являющимся таможенным представителем, либо указанным в сертификате открытого ключа. Электронный документ в таможенный орган не направлен. Представьте электронный документ с корректно заполненными сведениями.
100	Общая ошибка.
101	Ошибка авторизации (отсутствует заголовок UserId)
102	Отсутствует параметр запроса
103	Недопустимое значение параметра
104	Запись не найдена
105	Ошибка при разборе документа

Получение кода состояния HTTP отличного от 200 необходимо интерпретировать как неуспешный запрос.

5. Описание порядка взаимодействия

5.1. Сценарий информационного обмена при передаче ЭТД

Сценарий взаимодействия между ИСЗЛ и ИСТО через ОАИС при передаче ЭТД включает в себя следующие операции:

- подача документа от ИСЗЛ в ОАИС;
- проверка при приеме документа в ОАИС на обязательное заверение документа ЭЦП.
- проверка при приеме в ОАИС документа на полномочия лица на представление документа.

- передача по запросу сообщений от ОАИС в ИСЗЛ (отказ в передаче документа в ИСТО по причине не заверения документа ЭЦП/по причине отсутствия полномочий на представление документа);
- передача документа от ОАИС в ИСТО;
- обработка документа в ИСТО (регистрация/отказ в регистрации документа; результаты рассмотрения документа; формирование сведений о результатах рассмотрения документа).
- передача сообщений, сформированных ИСТО, в ОАИС;
- получение ИСЗЛ сообщений, сформированных ИСТО, из ОАИС.

5.1.1. Формирование и подписание документа в ИСЗЛ

5.1.1.1. ИСЗЛ формирует документ в формате XML и в соответствии со структурой, утвержденной решением Коллегии ЕЭК, действующим на дату подачи документа и опубликованном на официальном сайте Евразийской экономической комиссии.

5.1.2. Подача документа в ОАИС

5.1.2.1. ИСЗЛ отправляет сформированный документ, вызвав метод POST /request описанный в разделе 3.1, указав при этом:

- параметры для идентификации пользователя согласно разделу 2;
- заголовок «Content-Type: application/xml»;
- код таможенного органа, в который передается документ;
- GUID отправляемого файла, сформированный в ИСЗЛ.

5.1.2.2. При возникновении ошибки принятия передаваемого в ОАИС документа будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.2.3. Перед сохранением документа в ОАИС будет осуществляться обработка документа в соответствии с пунктами 5.1.2.4 - 5.1.2.5.

5.1.2.4. ОАИС осуществляет проверку документа на обязательное заверение ЭЦП.

5.1.2.5. В ОАИС осуществляется дополнительная проверка лица, представляющего документ в таможенный орган, на полномочия по его представлению.

5.1.2.6. В случае не прохождения любого контроля из пунктов 5.1.2.4 - 5.1.2.5 будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием причины. Структура этого JSON приведена в разделе 4.

5.1.2.7. При успешном прохождении предусмотренных пунктами 5.1.2.4 - 5.1.2.5 контролей в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться информация о принятом документе (запросе) в формате JSON (описание полей возвращаемого сообщения приведено в таблице 3.2). Код статуса запроса (status_id) для данного документа будет изменен на 0.

5.1.3. Передача и обработка документа в ИСТО. Передача сообщений, сформированных ИСТО, в ОАИС

5.1.3.1. ОАИС, при успешном принятии документа от ИСЗЛ, передает данный документ в ИСТО.

5.1.3.2. В случае неуспешного принятия ИСТО документа, ОАИС изменяет код статуса запроса (`status_id`) на 9.

5.1.3.3. В случае успешного принятия ИСТО документа, в ОАИС изменяет код статуса запроса (`status_id`) на 1.

5.1.3.4. ИСТО, в случае успешного прохождения входного и структурного контроля, передает в ОАИС документ «Извещение о принятии документа в ИСТО» (`DocumentAcceptanceNotice`) и меняет код статуса запроса (`status_id`) на 3.

Документ `DocumentAcceptanceNotice` заполняется следующим образом:

- поле `DocumentID` содержит уникальный идентификатор документа;
- поле `DateAccepted` содержит дату принятия документа в ИСТО;

Структура сообщения `DocumentAcceptanceNotice` приведена в таблицах Б.6, Б.7 Приложения Б.

5.1.3.5. ИСТО, при наличии оснований для отказа в принятии декларации, передает в ОАИС документ «Извещение об отказе в принятии документа в ИСТО» (`DocumentRejectionNotice`) и меняет код статуса запроса (`status_id`) на 2.

Документ `DocumentRejectionNotice` должен заполняться следующим образом:

- поле `DocumentID` содержит уникальный идентификатор документа;
- поле `DateRejected` содержит дату отказа в принятии документа в ИСТО;
- поле `ReasonCode` элемента `RejectionReason` содержит код причины отказа;
- поле `Description` элемента `RejectionReason` содержит текстовое описание причины отказа.

Структура сообщения `DocumentRejectionNotice` приведена в таблицах Б.4, Б.5 Приложения Б.

5.1.3.6. В случае отказа в регистрации декларации, ИСТО направляет в ОАИС документ «Извещение об отказе в регистрации» (`DocumentReturnNotice`) и меняет код статуса запроса (`status_id`) на 11.

Документ `DocumentReturnNotice` заполняется следующим образом:

- поле `DocumentID` содержит уникальный идентификатор документа, которому отказано в регистрации;
- поле `DateReturned` содержит дату отказа в регистрации;
- поле `ReturnReason` содержит текстовое описание причины отказа.

В документе может присутствовать элемент `ControlLog` (заполняется в соответствии с таблицами: Б.1. - Б.3. приложения Б), содержащий сведения из протокола форматно-логического контроля, поля которого заполняются следующим образом:

- поле `ControlDate` содержит дату проведения форматно-логического контроля;
- поле `EntryCount` содержит количество записей в протоколе ФЛК;
- элемент `Entries` содержит коллекцию элементов `Entry`, каждый из которых содержит следующую информацию:
 - поле `Type` содержит тип записи (0 – ошибка, 1- предупреждение, 2- информация);
 - поле `Section` содержит наименование графы, к которой относится запись;
 - поле `Field` содержит наименование поля, к которому относится запись;
 - поле `Code` содержит код ошибки;
 - поле `SubCode` содержит подкод ошибки;
 - поле `Text` содержит текстовое описание ошибки.

Сообщение `DocumentReturnNotice` заполняется в соответствии с таблицами Б.10, Б.11 Приложения Б.

5.1.3.7. В случае принятия решения о регистрации, ИСТО направляет в ОАИС документ «Извещение о номере регистрации документа» (`DocumentRegistrationNotice`), при этом ОАИС меняет код статуса запроса (`status_id`) на 5.

Документ `DocumentRegistrationNotice` заполняется следующим образом:

- поле `DocumentID` содержит уникальный идентификатор документа, которому присвоен регистрационный номер;
- поле `DateRegistered` содержит дату регистрации документа;
- поле `RegistrationNumber` содержит присвоенный документу регистрационный номер.

В документе может присутствовать элемент `ControlLog`, описание заполнения которого приведено в п. 5.1.3.6.

Сообщение `DocumentRegistrationNotice` заполняется в соответствии с таблицами Б.8, Б.9 Приложения Б.

5.1.4. Получение ИСЗЛ таможенных сообщений из ОАИС

5.1.4.1. Получение идентификатора запроса.

5.1.4.1.1. Для получения списка таможенных сообщений, переданных от ИСТО в ОАИС и привязанных к конкретному запросу (переданному от ИСЗЛ в ОАИС), необходим идентификатор этого запроса. Данный идентификатор запроса можно получить следующими способами:

- из поля «`id`» в ответном JSON при успешной отправке ЭТД в ОАИС;
- из списка запросов пользователя.

5.1.4.1.2. Для получения списка запросов пользователя ИСЗЛ необходимо вызвать метод `GET /requests` согласно разделу 3.2 (с параметрами для идентификации пользователя согласно разделу 2).

5.1.4.1.3. При успешном вызове метода в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться список запросов пользователя в

формате JSON. Описание полей возвращаемого сообщения приведено в таблице 3.5. Идентификатор запроса будет содержаться в поле «id».

5.1.4.1.4. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.1.5. При получении статуса ответа отличного от 200 операцию считать неуспешной.

5.1.4.2. Получение списка сообщений, связанных с запросом.

5.1.4.2.1. ИСЗЛ, путем вызова метода GET /files, описанного в разделе 3.4, получает список сообщений, связанных с запросом. При вызове метода указываются:

- параметры для идентификации пользователя согласно разделу 2;
- идентификатор запроса, полученный в п. 5.1.4.1.

5.1.4.2.2. При успешном вызове метода в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться в формате JSON список сообщений, связанных с запросом. Описание полей возвращаемого сообщения приведено в таблице 3.14. Идентификатор сообщения содержится в поле «ln_id», тип документа содержится в поле «ln_type». Список значений поля «ln_type» приведен в таблице 5.1.

Таблица 5.1

Тип сообщения	Наименование сообщения
0	Исходный документ
2	Извещение об отказе в принятии документа в ИСТО
3	Извещение о принятии документа в ИСТО
15	Извещение об отказе в регистрации документа
5	Извещение о номере регистрации документа

5.1.4.2.3. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.2.4. При получении статуса ответа отличного от 200 операцию считать неуспешной.

5.1.4.3. Получение сообщения, связанного с запросом.

5.1.4.3.1. ИСЗЛ, путем вызова метода GET /file, описанного в разделе 3.5, получает содержимое сообщения, связанного с запросом. При вызове метода указываются:

- параметры для идентификации пользователя согласно разделу 2;
- идентификатор сообщения, полученный в п. 5.1.4.2.2.

5.1.4.3.2. При успешном вызове метода в HTTP ответе будет возвращен статус 200, в заголовке указан «Content-Type: application/xml», а в теле ответа будет содержимое запрашиваемого таможенного сообщения в формате XML.

5.1.4.3.3. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.3.4. При получении статуса ответа отличного от 200 операцию считать неуспешной.

6. Правила использования ЭЦП при подписании ЭТД

6.1. Общие положения

Для формирования ЭЦП используются алгоритмы и средства криптографической защиты информации (СКЗИ), разрешенные к применению в таможенных органах Республики Беларусь.

Используемые алгоритмы и СКЗИ должны соответствовать следующим документам:

- СТБ 34.101.45-2013 «Информационная технология и безопасность. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых»;

- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

Порядок формирования и использования ЭЦП для обеспечения достоверности электронного документа определяется в соответствии со спецификацией W3C «XML-Signature Syntax and Processing» (подпись XML) и нормативными документами Республиканского удостоверяющего центра ГосСУОК.

6.2. Порядок формирования ЭЦП

6.2.1. Создание элемента Signature, который содержит четыре дочерних элемента: SignedInfo, SignatureValue, KeyInfo и Object. Этот элемент является единственным обертывающим элементом для других тегов цифровой подписи XML. Атрибут ID данного элемента должен иметь значение атрибута ID первого из подписываемых элементов, с добавлением перед ним символов «SID-».

6.2.2. Создание дочерних элементов элемента Object.

Элемент предназначен для указания дополнительных данных, которые должны быть подписаны. Данным документом определяется использование данного элемента для создания метки времени.

Атрибут ID элемента Object должен содержать значение атрибута ID элемента Signature с добавлением перед ним символа «T».

Элемент Object должен содержать элемент SignatureProperties, который в свою очередь должен содержать элемент SignatureProperty, атрибут Target которого должен быть равен значению атрибута ID элемента Signature.

Элемент `SignatureProperty` должен содержать элемент `SigningTime` типа `xs:datetime` из пространства имен «<http://lab119.net/STBCrypt>», значение которого должно быть равно времени формирования ЭЦП в UTC.

6.2.3. Создание дочерних элементов элемента `SignedInfo`.

Элемент `CanonicalizationMethod` – обязательный элемент, который идентифицирует алгоритм канонизации, применяемой к элементу `SignedInfo` до создания подписи. Атрибут `Algorithm` данного элемента должен иметь значение «<http://www.w3.org/TR/2001/REC-xml-c14n-20010315>».

Элемент `SignatureMethod` – указывает алгоритм, используемый для создания подписи. Атрибут `Algorithm` данного элемента должен иметь значение «<http://www.w3.org/2001/04/xmldsig-more#STB34101312011-STB34101452013>».

Элементы `Reference` – определяет данные, которые будут подписаны, а также преобразования, которые необходимо применить к этим данным перед вычислением значения подписи.

Каждый элемент `Reference` должен содержать следующую информацию:

- атрибут `URI` – ссылка на подписываемый элемент внутри данного XML документа. Значение атрибута должно быть равно значению атрибута `ID` подписываемого элемента с добавлением перед ним символа «#»;

- элемент `Transforms` – определяет преобразования данных до их подписания. Должен содержать единственное преобразование (элемент `Transform`) – приведение к каноническому виду (атрибут `Algorithm` со значением «<http://www.w3.org/TR/2001/REC-xml-c14n-20010315>»);

- элемент `DigestMethod` – указывает алгоритм, используемый для создания дайджеста. Атрибут `Algorithm` данного элемента должен иметь значение «<http://www.w3.org/2001/04/xmldsig-more#STB34101312011>»;

- элемент `DigestValue` – значение дайджеста. Должен содержать значение функции хеширования СТБ 34.101.31-2011 в кодировке `base64`, входными данными для которой является содержимое элемента (включая открывающий и закрывающий тэги), ссылка на который указана в атрибуте `URI` элемента `Reference`, приведенное к каноническому виду.

Количество элементов `Reference` должно быть равно количеству подписываемых элементов. Дополнительно каждая подпись должна содержать элемент `Reference` со значением атрибута `URI` равным значению атрибута `ID` элемента `Object` с добавлением перед ним символа «#» (ссылка на метку времени).

6.2.4. Создание дочерних элементов элемента `KeyInfo`.

Элемент `KeyInfo` должен содержать элемент `X509Data`, содержимое которого содержит информацию о подписавшем. Элемент `X509Data` должен содержать следующие элементы:

- элемент `X509IssuerSerial` – информация об издателе и серийном номере сертификата подписавшего. Данное сочетание позволяет однозначно идентифицировать любой сертификат открытого ключа. Элемент `X509IssuerSerial` должен содержать элемент `X509IssuerName`, значение которого должно содержать имя издателя сертификата, и элемент

X509SerialNumber, значение которого должно содержать серийный номер сертификата в десятичной системе счисления;

- элемент X509SKI – идентификатор открытого ключа субъекта. Должен содержать первые 20 байт идентификатора открытого ключа субъекта, вычисленного в соответствии с СТБ 34.101.45-2013, в кодировке base64;

- элемент X509Certificate – сертификат подписавшего. Должен содержать бинарное представление сертификата подписавшего в кодировке base64;

- элемент X509CRL (необязательный) – список отозванных сертификатов. Может содержать бинарное представление действующего на момент подписи списка отозванных сертификатов в кодировке base64.

6.2.5 Вычисление значение элемента SignatureValue.

Сформированный в п. 6.2.3 элемент SignedInfo приводится к каноническому виду, после чего обрабатывается функцией хеширования СТБ 34.101.31-2011. Полученное значение дайджеста передается на вход процедуре выработки ЭЦП СТБ 34.101.45-2013 с использованием личного ключа подписи, соответствующего открытому ключу подписи, содержащемуся в сертификате, указанному в элементе KeyInfo. Полученное значение ЭЦП помещается в элемент SignatureValue в кодировке base64.

6.3. Порядок проверки ЭЦП

6.3.1. Проверка значений дайджестов в элементах Reference.

Для каждого элемента Reference, содержащихся в элементе References элемента SignedInfo необходимо вычислить значение функции хеширования СТБ 34.101.31-2011, входными данными для которой является содержимое элемента (включая открывающий и закрывающий тэги), ссылка на который указана в атрибуте URI, приведенное к каноническому виду. Если вычисленное значение не совпадает с указанным в элементе DigestValue элемента Reference, принимается решение о недействительности ЭЦП.

6.3.2. Получение информации об открытом ключе подписи из элемента KeyInfo.

Сертификат открытого ключа подписи может быть получен:

- из элемента X509Certificate;
- из локального или сетевого хранилища сертификатов по информации об издателе и серийном номере сертификата;

- из локального или сетевого хранилища сертификатов по идентификатору открытого ключа субъекта.

Если вышеописанные способы не позволили получить сертификат подписавшего, принимается решение о невозможности проверки ЭЦП, и, следовательно, ее недействительности.

6.3.3. Проверка действительности сертификата на момент подписи.

Должны быть выполнены следующие проверки:

- построена цепочка сертификатов до сертификата корневого удостоверяющего центра ГосСУОК;

- проверен срок действия каждого из сертификатов в цепочке, при этом время подписи должно входить в срок действия каждого из сертификатов;
- проверена ЭЦП каждого сертификата с использованием открытого ключа подписи, указанного в сертификате издателя;
- проверен статус отзыва каждого из сертификатов в цепочке путем анализа соответствующих списков отозванных сертификатов на предмет нахождения в них проверяемых сертификатов, либо иными способами, определенными соответствующим УЦ (при этом может быть использован список отозванных сертификатов, указанный в элементе X509CRL).

При отрицательном результате любой из вышеуказанных проверок принимается решение о недействительности сертификата подписавшего на момент постановки подписи, и, следовательно, недействительности ЭЦП.

Время постановки подписи должно быть получено из элемента SigningTime. При невозможности определить время постановки подписи, в проверках используется текущее время.

6.3.4. Проверка значения ЭЦП указанного в элементе SignatureValue.

Элемент SignedInfo приводится к каноническому виду, после чего обрабатывается функцией хеширования СТБ 34.101.31-2011. Полученное значение дайджеста передается на вход процедуре проверки ЭЦП СТБ 34.101.45-2013 с использованием открытого ключа подписи, содержащегося в сертификате. При отрицательном результате процедуры проверки ЭЦП принимается решение о недействительности ЭЦП, при положительном – принимается решение о том, что ЭЦП действительна.

6.4. Использование ЭЦП в прикладных электронных документах

6.4.1. Общие положения.

ЭЦП на документ накладывает лицо, сформировавшее документ. При этом декларант накладывает ЭЦП только на поля, заполняемые декларантом.

Сотрудник таможенного органа при оформлении документа накладывает ЭЦП как на поля, заполняемые декларантом (включая данные об ЭЦП декларанта), так и на поля, заполняемые сотрудником таможенного органа.

6.4.2. При формировании документа декларантом, лицо формирующее документ обязано подписать элемент Declarant. Сформированный элемент Signature должен быть помещен в корневой элемент ТМРА документа. Атрибут ID элемента Signature должен иметь значение равное значению атрибута ID элемента Declarant с добавлением перед ним символов «SID-».

6.4.3. При формировании электронного свидетельства, сотрудник таможенного органа накладывает ЭЦП на следующие элементы (в порядке следования):

- элемент Customs;
- элемент Declarant;
- элемент Signature, сформированный декларантом.

Сформированный элемент Signature должен быть помещен в корневой элемент ТМРА документа. Атрибут ID элемента Signature должен иметь значение равное значению атрибута ID элемента Custom с добавлением перед ним символов «SID-».

6.5. Нормативные документы по ЭЦП

Используются следующие нормативные документы по ЭЦП:

- СТБ 34.101.45-2013 «Информационная технология и безопасность. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых»;

- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

- СТБ 34.101.19-2012 «Информационные технологии и безопасность. Форматы сертификатов и списков отозванных сертификатов инфраструктуры открытых ключей»;

- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

- «Extensible Markup Language (XML) 1.0» <http://www.w3.org/TR/xml/>;

- «Namespaces in XML» <http://www.w3.org/TR/REC-xml-names>;

- «XML Schema Part 1:Structures» <http://www.w3.org/TR/xmlschema-1>;

- «XML Schema Part 2:Datatypes» <http://www.w3.org/TR/xmlschema-2>;

- «Web Services Description Language (WSDL) 1.1» <http://www.w3.org/TR/wsdl>;

- «Simple Object Access Protocol (SOAP) 1.2» <http://www.w3.org/TR/soap12-part0>, <http://www.w3.org/TR/soap12-part1>, <http://www.w3.org/TR/soap12-part2>;

- «Web Services Addressing (WS-Addressing) 1.0» <http://www.w3.org/Submission/ws-addressing/>;

- «XML Signature Syntax and Processing (Second Edition) » <http://www.w3.org/TR/xmlsig-core>;

- «Canonical XML 1.0» <http://www.w3.org/TR/xml-c14n>.

7. Требования при взаимодействии с ОАИС

Требования приведены в Порядке оказания электронных услуг ОАИС, размещенном на официальном сайте республиканского унитарного предприятия «Национальный центр электронных услуг» (https://nces.by/wp-content/uploads/poradok_oais.pdf).

XML схема технологических электронных документов

```

<xs:schema xmlns:tns="http://gtk.gov.by/CustomsService"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://gtk.gov.by/CustomsService"
elementFormDefault="qualified">
  <xs:complexType name="ControlLog">
    <xs:sequence>
      <xs:element name="ControlDate" type="xs:dateTime"/>
      <xs:element name="EntryCount" type="xs:int"/>
      <xs:element name="Entries"
type="tns:ArrayOfControlLogEntry"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfControlLogEntry">
    <xs:sequence>
      <xs:element name="Entry" type="tns:ControlLogEntry"
nillable="true" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ControlLogEntry">
    <xs:sequence>
      <xs:element name="Type" type="xs:int"/>
      <xs:element name="Section" type="xs:string"
minOccurs="0"/>
      <xs:element name="Field" type="xs:string" minOccurs="0"/>
      <xs:element name="Code" type="xs:string" minOccurs="0"/>
      <xs:element name="SubCode" type="xs:string"
minOccurs="0"/>
      <xs:element name="Text" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="DocumentRejectionNotice"
type="tns:DocumentRejectionNotice" nillable="true"/>
  <xs:complexType name="DocumentRejectionNotice">
    <xs:sequence>
      <xs:element name="NoticeInfo"
type="tns:DocumentRejectionNoticeInfo"/>
    </xs:sequence>
  </xs:complexType>

```



```

<xs:complexType name="DocumentRejectionNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRejected" type="xs:dateTime"/>
    <xs:element name="RejectionReason" type="tns:Reason"/>
    <xs:element name="ControlLog" type="tns:ControlLog"
minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:complexType name="Reason">
  <xs:sequence>
    <xs:element name="ReasonCode" type="xs:string"/>
    <xs:element name="Description" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="DocumentAcceptanceNotice"
type="tns:DocumentAcceptanceNotice" nillable="true"/>
<xs:complexType name="DocumentAcceptanceNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo"
type="tns:DocumentAcceptanceNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentAcceptanceNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateAccepted" type="xs:dateTime"/>
    <xs:element name="AcceptanceNumber" type="xs:string"
minOccurs="0"/>
    <xs:element name="ControlLog" type="tns:ControlLog"
minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentRegistrationNotice"
type="tns:DocumentRegistrationNotice" nillable="true"/>
<xs:complexType name="DocumentRegistrationNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo"
type="tns:DocumentRegistrationNoticeInfo"/>
  </xs:sequence>

```

```

</xs:complexType>
<xs:complexType name="DocumentRegistrationNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
    <xs:element name="RegistrationNumber" type="xs:string"/>
    <xs:element name="ControlLog" type="tns:ControlLog"
minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentReturnNotice"
type="tns:DocumentReturnNotice" nillable="true"/>
<xs:complexType name="DocumentReturnNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo"
type="tns:DocumentReturnNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentReturnNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateReturned" type="xs:dateTime"/>
    <xs:element name="ReturnReason"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
</xs:schema>

```

Структуры электронных таможенных сообщений (документов)

Таблица Б.1. Структура типа данных ControlLog.

Элемент		Описание элемента	Тип	Мн.
1	ControlDate	Дата проведения ФЛК	xs:datetime	[1]
2	EntryCount	Количество ошибок ФЛК	xs:integer	[1]
3	Entries	Перечень ошибок ФЛК	ArrayOfControlLogEntry	[1]

Таблица Б.2. Структура типа данных ArrayOfControlLogEntry.

Элемент		Описание элемента	Тип	Мн.
1	Entry	Информация об ошибке	ControlLogEntry	[0..n]

Таблица Б.3. Структура типа данных ControlLogEntry.

Элемент		Описание элемента	Тип	Мн.
DocumentNonEntryNotice		Корневой элемент		[1]
1	Type	Тип ошибки (0 – ошибка, 1 – предупреждение, 2 – информация)	xs:int	[1]
2	Section	Графа, в которой допущена ошибка	xs:string	[0..1]
3	Field	Поле, в котором допущена ошибка	xs:string	[0..1]
4	Code	Код ошибки	xs:string	[0..1]
5	SubCode	Подкод ошибки	xs:string	[0..1]
6	Text	Текстовое описание ошибки	xs:string	[1]

Таблица Б.4. Структура элемента DocumentRejectionNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentRejectionNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.5. Структура типа данных DocumentRejectionNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateRejected	Дата отказа в принятии документа в ИСТО	xs:datetime	[1]
3	RejectionReason	Причина отказа	Reason	[1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Структура типа данных Reason.

Элемент		Описание элемента	Тип	Мн.
DocumentListRequest		Корневой элемент		[1]
1	ReasonCode	Код причины (ошибки)	xs:string	[1]
2	Description	Описание причины (ошибки)	xs:string	[1]

Таблица Б.6. Структура элемента DocumentAcceptanceNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentAcceptanceNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.7. Структура типа данных DocumentAcceptanceNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateAccepted	Дата принятия документа в ИСТО	xs:datetime	[1]
3	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Таблица Б.8. Структура элемента DocumentRegistrationNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentRegistrationNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.9. Структура типа данных DocumentRegistrationNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateRegistered	Дата регистрации документа	xs:datetime	[1]
3	RegistrationNumber	Регистрационный номер документа	xs:string	[1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Таблица Б.10. Структура элемента DocumentReturnNotice

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentReturnNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.11. Структура типа данных DocumentReturnNoticeInfo

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateReturned	Дата отказа в регистрации документа	xs:datetime	[1]
3	ReturnReason	Причина отказа	xs:string	[1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]