ЦЕНТРАЛЬНЫЙ БАНК РОССИЙСКОЙ ФЕДЕРАЦИИ(БАНК РОССИИ)

ВНЕШНИЙ ПОРТАЛ БАНКА РОССИИ

Инструкция по отправке в Банк России через

Личный кабинет участника информационного обмена

заявлений по процедурам допуска (ФЭД)

с использованием REST-API сервиса

Участник информационного обмена имеет возможность отправить документы по процедурам допуска на финансовый рынок посредством REST-API сервиса или со стандартной формой прикрепления файла без заполнения формы.

В случае использования REST-API сервиса, участнику необходимо:

- 1. Подготовить пакет электронных документов, состоящий из:
 - а. файла "form.xml", содержащего заполненную форму отчетности;
 - b. сопроводительных файлов согласно описи в форме отчетности;
 - с. файлов отсоединенной подписи к каждому из вышеуказанных файлов.
- 2. Посредством запроса POST сформировать электронное сообщение по задаче Zadacha 77».
- 3. Создать по каждому файлу пакета электронных документов сессию загрузки запросом POST */createUploadSession, после чего загрузить файлы в личный кабинет запросами PUT.
- 4. После загрузки всех файлов, следует выполнить отправку электронного сообщения запросом POST.

Описания структуры запросов и примеры сформированных запросов доступны на сайте Банка России в разделе «Личный кабинет участника информационного обмена / Инструкции и иная информация о технологии подготовки и направления электронных документов в Банк России / REST-API сервис».

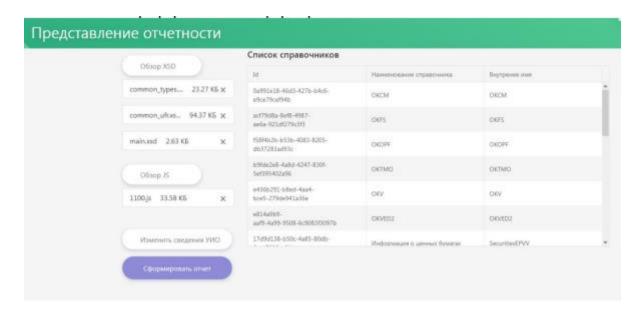
Ниже представлена подробная инструкция по отправке документов по процедурам допуска с помощью REST-API сервиса по форме 1100 и форме 1210.

Пример 1: форма 1100

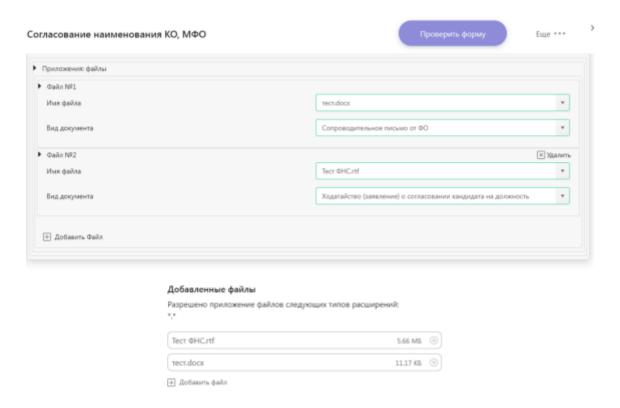
Входные данные: подготовленные файлы «Тест ФНС.rtf», «тест.docx», «form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml», подписи к ним, их зашифрованные версии, схемы и скрипт для формы 1100.

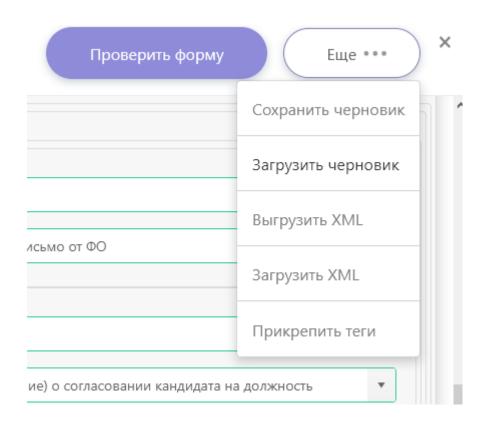
Процесс:

1. Открыть эмулятор ЭФ https://portal5test.cbr.ru/report/, загрузить XSD-схемы и Java-script формы 1100*. Сформировать отчет.



2. Заполнить форму, приложить сопроводительные файлы. Выгрузить XML версию заполненной формы.





- 3. Подписать и зашифровать все файлы, включая файл электронной формы. В итоге у участника должны быть пары «Зашифрованный файл» -«Подпись к нему». В тестовом примере:
- a. form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc
- b. form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.sig
- с. Тест ФНС.rtf.enc
- d. Tecτ ΦHC.rtf.sig
- е. тест.docx.enc
- f. тест.docx.sig
 - 4. Для формирования электронного сообщения следует направить запрос POST на адрес https://portal5.cbr.ru/back/rapi2/messages*, описывающий прилагаемые файлы. Запрос для тестового примера в формате CURL:

```
curl --request POST \
--url https://portal5.cbr.ru/back/rapi2/messages \
--header 'Authorization: Basic
MjJTaWRvQUUwMDc3MDIyMzUxMzM6bU90b3JhdmlhNg==' \
--header 'Content-Type: application/json' \
--data '{
```

```
"Task": "Zadacha_77",
"Files": [
"Name": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc",
"Encrypted": true,
"RepositoryType": "http",
"Size": "1638"
},
"Name": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.sig",
"Encrypted": "0",
"SignedFile": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc",
"RepositoryType": "http",
"Size": "3466"
},
"Name": "Tecτ ΦHC.rtf.enc",
"Encrypted": true,
"RepositoryType": "http",
"Size": "5963784"
},
"Name": "Tecτ ΦHC.rtf.enc.sig",
"Encrypted": "0",
"SignedFile": "Τecτ ΦHC.rtf.enc",
"RepositoryType": "http",
"Size": "3334"
},
"Name": "rect.docx.enc",
"Encrypted": true,
"RepositoryType": "http",
"Size": "12162"
},
```

```
{
"Name": "Tect.docx.enc.sig",
"Encrypted": "0",
"SignedFile": "Tect.docx.enc",
"RepositoryType": "http",
"Size": "3322"
}
]
}
```

```
"Task": "Zadacha_77",
    "Name": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc",
    "Encrypted": true,
    "RepositoryType": "http",
    "Size": "1638"
 },
   "Name": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.sig",
   "Encrypted": "0",
    "SignedFile": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc",
    "RepositoryType": "http",
    "Size": "3466"
    "Name": "Tect OHC.rtf.enc",
    "Encrypted": true,
"RepositoryType": "http",
    "Size": "5963784"
 },
   "Name": "Tect OHC.rtf.enc.sig",
   "Encrypted": "0",
   "SignedFile": "Tect OHC.rtf.enc",
   "RepositoryType": "http",
   "Size": "3334"
    "Name": "recr.docx.enc",
   "Encrypted": true,
    "RepositoryType": "http",
    "Size": "12162"
 },
    "Name": "recr.docx.enc.sig",
    "Encrypted": "0",
    "SignedFile": "tect.docx.enc",
    "RepositoryType": "http",
    "Size": "3322"
}
```

5. В ответном сообщении будут получены ID сообщения и файлов:

```
"Id": "6e797df4-def1-4d84-90f0-0a9daa98c0e5",
  "CorrelationId": null,
  "GroupId": null,
  "Type": "outbox",
  "CreationDate": "2021-02-08T13:27:47.3077553Z",
  "UpdatedDate": null,
  "Status": "draft",
  "TaskName": "Zadacha_77",
"RegNumber": null,
  "TotalSize": 5987706,
  "Sender": {
    "Inn": "7702235133",
    "0grn": "1037700013020",
    "Bik": "124556789",
    "RegNum": "78751",
    "DivisionCode": "0000"
  },
  "Files": [
      "Id": "4d9fb13a-4da2-4a36-95a6-aa6becfc5f64",
      "Name": "form-c907ad00-6219-11eb-a8d7-a12eac5657fc.xml.enc",
      "Description": null,
      "Encrypted": true,
      "SignedFile": null,
      "Size": 1638,
       "RepositoryInfo": [
           "RepositoryType": "http",
"Host": "portal5.cbr.ru",

"Port": 81,

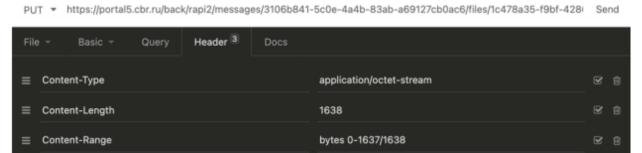
"Path": "rapi2/messages/6e797df4-def1-4d84-90f0-0a9daa98c0e5/files/4d9fb13a-
4da2-4a36-95a6-aa6becfc5f64/download"
```

6. По каждому из файлов необходимо выполнить запрос сессии загрузки – запрос POST на следующий адрес:

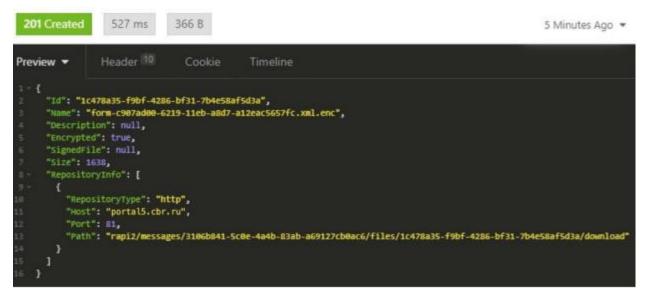
https://portal5.cbr.ru/back/rapi2/messages/MESSAGE_ID/files/FILE_ID/createUploadSession где MESSAGE_ID и FILE_ID берутся из ответа сервера в п.5. В ответ сервер вышлет адрес, по которому необходимо загрузить файл.



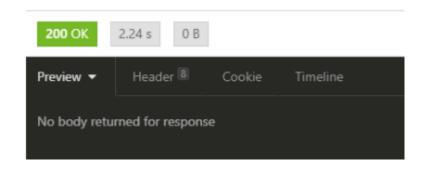
7. По каждой из открытых сессий загрузки необходимо выполнить запрос PUT на полученный в п.6 адрес, поместив в тело запроса файл и проставив метаданные: — Content-Length — размер файла в байтах (в примере — 1638); — Content-Range — размер передаваемой части файла в байтах (в примере — bytes 0-1637/1638); — Content-Type — тип данных (в примере — application/octet-stream).



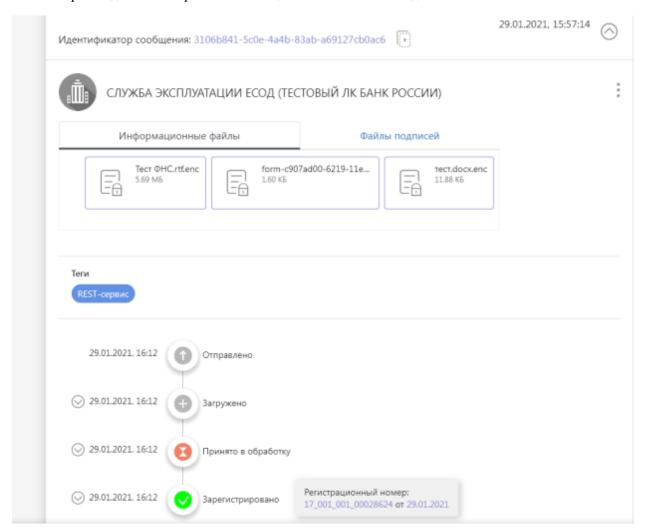
В случае успешного выполнения запроса от сервера будет получен статус 201 Created и описание файла:



8. После загрузки всех файлов в личный кабинет необходимо выполнить запрос POST по адресу https://portal5.cbr.ru/back/rapi2/messages/MESSAGE_ID, где MESSAGE_ID был получен в п.5. В случае успешного выполнения запроса от сервера будет получен статус 200 OK.



9. Прохождение электронного сообщения возможно отследить в Личном кабинете:

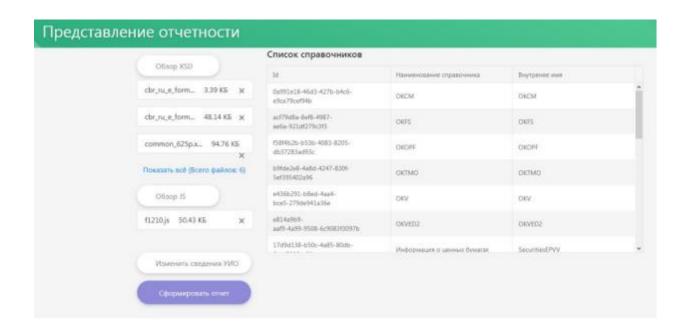


Пример 2: форма 1210

Входные данные: подготовленные файлы «тест.docx», «form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml», подписи к ним, их зашифрованные версии, схемы и скрипт для формы 1210.

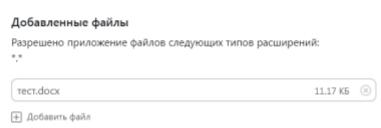
Процесс:

1. Открыть эмулятор ЭФ https://portal5test.cbr.ru/report/, загрузить XSD-схемы и Java-script формы 1210*. Сформировать отчет.



2. Заполнить форму, приложить сопроводительные файлы. Выгрузить XML версию заполненной формы. В тестовом варианте: сравнить полученный файл с «form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml»:





3. Подписать и зашифровать все файлы, включая файл электронной формы. В итоге у участника должны быть пары «Зашифрованный файл» -«Подпись к нему». В тестовом примере:

```
a. form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc
b. form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.sig
с. тест.docx.enc
d. тест.docx.sig
   4. Для формирования электронного сообщения следует направить запрос POST*: curl --
       request POST \
--url https://portal5.cbr.ru/back/rapi2/messages \
--header 'Authorization: Basic
MjJTaWRvQUUwMDc3MDIyMzUxMzM6bU90b3JhdmlhNg=='\
--header 'Content-Type: application/json' \
--data '{
"Task": "Zadacha_77",
"Files": [
"Name": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc",
"Encrypted": true,
"RepositoryType": "http",
"Size": "3947"
},
"Name": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.sig",
"Encrypted": "0",
"SignedFile": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc",
"RepositoryType": "http",
"Size": "3466"
},
"Name": "rect.docx.enc",
```

"Encrypted": true,

```
"RepositoryType": "http",
"Size": "12162"
},
{
"Name": "rect.docx.enc.sig",
"Encrypted": "0",
"SignedFile": "rect.docx.enc",
"RepositoryType": "http",
"Size": "3322"
}
]
```

```
Header 2
JSON ▼
              Auth =
                           Query
                                                      Docs
      "Task": "Zadacha_77",
           "Name": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc",
          "Encrypted": true,
"RepositoryType": "http",
           "Size": "3947"
          "Name": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.sig",
          "Encrypted": "0",
           "SignedFile": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc",
          "RepositoryType": "http",
           "Size": "3466"
           "Name": "recr.docx.enc",
          "Encrypted": true,
"RepositoryType": "http",
           "Size": "12162"
         },
           "Name": "recr.docx.enc.sig",
          "Encrypted": "0",
"SignedFile": "Tect.docx.enc",
           "RepositoryType": "http",
           "Size": "3322"
```

5. В ответном сообщении будут получены ID сообщения и файлов:

200 OK 352 ms 1877 B Header 10 Preview ▼ Cookie Timeline "Id": "99d282f2-c893-440a-85f4-549d72796d1f", "CorrelationId": null, "GroupId": null,
"Type": "outbox",
"Title": null, "Text": null, "CreationDate": "2021-02-08T13:32:48.1721164Z", "UpdatedDate": null, "Status": "draft", "TaskName": "Zadacha_77", "RegNumber": null, "TotalSize": 22897, "Sender": { "Inn": "7702235133", "Ogrn": "1037700013020", "Bik": "124556789", "RegNum": "78751", "DivisionCode": "0000" }, "Files": ["Id": "01b15631-7d51-43f5-9bd7-0c8ead648f05", "Name": "form-52804870-621b-11eb-8c68-bbde69f5a7e4.xml.enc", "Description": null, "Encrypted": true, "SignedFile": null, "Size": 3947, "RepositoryInfo": [

6. Шаги 6-9 аналогичны шагам из примера 1. После отправки прохождение сообщения возможно отследить в ЛК:

