



## Межбанковские расчеты

### Оглавление

- Термины и сокращения
- Инструкция по подключению
- Настройка защищённого канала взаимодействия
  - Формирование подписи тела сообщения
  - Выпуск сертификата
  - Создание VPN-туннеля
- Аутентификация
- Процесс межбанковских расчетов
  - Описание
  - Схема взаимодействия
- Статусы операций, коды ошибок
  - Статусы операции проверки верификации исходящего документа
  - Справочник кодов детализации ошибок
- Тестовые сценарии
- Описание эндпоинтов
  - Направить документ в Банк
  - Проверка верификации отправленного документа
  - Получить список документов от Банка
  - Получить входящий документ

## Термины и сокращения

Термин	Определение
Банк	КБ «Долинск» (АО)
Банк-партнер	Клиент Банка
МБР	Межбанковские расчеты

# Инструкция по подключению

## Шаг 1. Подпишите договор присоединения к правилам Платежного шлюза Банка

Обратитесь в Банк по любому каналу связи:

- Телефон: по РФ 8-800-200-45-75 бесплатно
- Почта: [web@bankdolinsk.ru](mailto:web@bankdolinsk.ru)

Менеджер Банка проконсультирует по подключению и вышлет пакет документов.

## Шаг 2. Организуйте защищенный канал

Для этого:

- Выпустите [сертификат](#). Предоставьте открытую часть сертификата.
- Получите от нас ключ и организуйте [VPN туннель](#).

## Шаг 3. Реализуйте интеграцию

Интеграция происходит в два этапа:

Первый этап:

- Получите от сотрудника банка набор данных для [подключения в тестовый контур](#).
- Реализуйте и пройдите [тестовые сценарии](#).
- Сообщите о завершении интеграции менеджеру банка и отправьте ему заполненный файл отчёта о тесте.

Второй этап:

- Сотрудник банка проверит выполненную интеграцию и отправит набор данных для подключения на промышленный контур.

## Настройка защищённого канала взаимодействия

Передача данных при взаимодействии с Банком осуществляется с использованием ГОСТ VPN-туннеля.

Для организации туннеля и подписи запросов используется **СКЗИ КриптоПро CSP 5.0** (рекомендуется использовать актуальную сертифицированную версию, на текущий момент это CSP 5.0 R3), список поддерживаемых операционных систем доступен по ссылке: <https://cryptopro.ru/products/csp/compare>.

Кроме **КриптоПро CSP** потребуются:

- **КриптоПро JCP (JCP 2.0)** – создание и проверка ЭЦП:  
<https://cryptopro.ru/products/csp/jcp> (требуется Java 7 Runtime Environment версии 7 и выше).
- **КриптоПро Stunnel** – предназначена для организации VPN-туннеля, для шифрования трафика: <https://cryptopro.ru/products/other/stunnel>

Конфигурация машины для подписи/шифрования выбирается с учетом используемой операционной системы + выбранного Java Runtime Environment. Дополнительная нагрузка, в моменте, напрямую зависит от плотности потока обрабатываемых данных.

## Формирование подписи тела сообщения

Подпись запросов к API осуществляется с помощью алгоритмов **ГОСТ Р 34.11-2012 256 бит** и **ГОСТ Р 34.10-2012 256 бит**.

Сформированная подпись кодируется в `base64` и передается в заголовке `Signature`.

Пример кода на языке Java, формирующего подпись для передаваемого в Банк файла:

```
String payload = objectMapper.writeValueAsString(body).getBytes(UTF_8);
Signature signer = Signature.getInstance(signAlgorithm, JCP.PROVIDER_NAME);
// signAlgorithm по умолчанию GOST3411_2012_256withGOST3410_2012_256

signer.initSign(privateKey);
// privateKey - ключ подписания, по которому с помощью сертификата будет
// проверена подпись

signer.update(payload); // payload - byte[] тело запроса
byte[] rawSign = signer.sign()
String b64Sign = Base64.getEncoder().encodeToString(rawSign)
// b64Sign - подпись в кодировке base64, это значение нужно передать в
// хедере Signature
```

## Выпуск сертификата

### ⚠ ВАЖНО!

- Дальнейшая инструкция предполагает использование ОС семейства Linux.
- При использовании ОС Windows просим использовать официальные инструкции для соответствующих действий.
- Выпуск сертификата желательно производить под тем же пользователем, под которым в дальнейшем предполагается запускать Stunnel.

После установки КриптоПро, JCP и Stunnel (входит в дистрибутивы КриптоПро для ОС семейства Linux) согласно официальным инструкциям можно приступать к выпуску сертификата. Для шифрования исходящего трафика и подписи запросов достаточно выпустить самоподписанный сертификат.

- Создадим переменную `$ALIAS`, в значение которой запишем название организации, для которой выпускаем сертификат (в примере – `MY_BANK`):

```
export ALIAS=MY_BANK
```

- Выпустим сертификат, для владельца (**-dn**) и издателя (**-issuer**) будем использовать одинаковые данные `"CN=MY_NAME, O=$ALIAS, L=MY_LOCATION, C=MY_COUNTRY"`:

```
/opt/cprosp/bin/amd64/csptest -minica -root -cont '\\.\HDIMAGE\'$ALIAS -provtype 80 -issuer "CN=MY_NAME, O=$ALIAS, L=MY_LOCATION, C=MY_COUNTRY" -dn "CN=MY_NAME, O=$ALIAS, L=MY_LOCATION, C=MY_COUNTRY" -until 365 -store uMy -certusage '1.3.6.1.5.5.8.2.2,1.3.6.1.5.5.7.3.2,1.3.6.1.5.5.7.3.1'
```

- Выгрузим публичную часть сертификата в файл `$ALIAS.cer` в текущей директории:

```
/opt/cprosp/bin/amd64/certmgr -export -store uMy -dest $ALIAS.cer
```

- Свяжем выгруженную публичную часть сертификата `$ALIAS.cer` с контейнером закрытого ключа `\\.\HDIMAGE\'$ALIAS`:

```
/opt/cprosp/bin/amd64/csptest -keyset -cont '\\.\HDIMAGE\'$ALIAS -keytype exchange -impcert $ALIAS.cer
```

- Проверим наличие сертификата в хранилище текущего пользователя и наличие связи с контейнером закрытого ключа (PrivateKey Link : Yes):

```
/opt/cprosp/bin/amd64/certmgr -list -store uMy
```

- Для просмотра информации о полученном сертификате, которая понадобится при заполнении заявлений:

```
/opt/cprosp/bin/amd64/certmgr -list -file $ALIAS.cer -verbose
```

### ДОПОЛНИТЕЛЬНЫЕ КОМАНДЫ

- Полный список всех доступных контейнеров:

```
/opt/cprosp/bin/amd64/csptest -keyset -enum_cont -fqcn -verifyc
```

- Просмотр информации о контейнере:

```
/opt/cprosp/bin/amd64/csptest -keyset -check -cont '\\.\HDIMAGE\  
<container_name>'
```

- Сделать пароль контейнера закрытого ключа пустым:

```
/opt/cprosp/bin/amd64/csptest -passwd -change "" -container "  
<container_name>"
```

- Просмотр информации о лицензии КриптоПро:

```
/opt/cprosp/sbin/amd64/cpconfig -license -view
```

## Создание VPN-туннеля

Для организации ГОСТ VPN-туннеля используется программа КриптоПро Stunnel `stunnel-msspi` (входит в дистрибутив КриптоПро CSP при использовании ОС семейства Linux).

Установить программу можно следующей командой:

```
# ./install.sh cproscsp-stunnel-msspi
```

Пример команды запуска Stunnel:

```
/opt/cproscsp/sbin/amd64/stunnel_thread <path_to_stunnel.conf >
```

где `stunnel.conf` – файл конфигурации Stunnel.

Для организации и сопровождения туннеля, в файле конфигурации достаточно указать настройки:

```
pid=/var/stunnel/stunnel.pid ;Файл для сохранения pid
output=/var/stunnel/log/stunnel.log ;Путь к лог-файлу stunnel
socket=l:TCP_NODELAY=1 ;Опции setsockopt() для сокета приема соединений, а
так же для локального и удаленного сокетов
socket=r:TCP_NODELAY=1
debug=7 ;Уровень логирования, можно снизить после отладки
[https]
client=yes ;Работа в режиме клиента
accept=127.0.0.1:1500 ;Адрес:порт, куда будет смотреть ваше приложение
connect=<АдресСервера>:443 ;Адрес подключения, выданный Банком
cert= /var/stunnel/$ALIAS.cer ;Путь к публичной части сертификата Банка-
Клиента (полученной в предыдущем пункте документации)
verify=2 ;Уровень проверки сертификата, при отладке можно выставить
значение "0"
```

Предоставленный Банком сертификат сервера, обычно `gateway-server.cer`, нужно установить в хранилище. При использовании самоподписанного сертификата (текущая версия) выполнить команды:

```
/opt/cprosp/bin/<архитектура>/certmgr -inst -file gateway-server.cer -  
store uMy  
/opt/cprosp/bin/<архитектура>/certmgr -inst -file gateway-server.cer -  
store CA  
/opt/cprosp/bin/<архитектура>/certmgr -inst -file gateway-server.cer -  
store ROOT
```

При использовании сертификата выпущенного официальным центром сертификации в хранилища ROOT и CA добавить соответствующие сертификаты центра сертификации, в личное – добавить свой сертификат.

Если в качестве адреса для подключения (<АдресСервера> в строке connect=<АдресСервера>:443 выше) Банк предоставил IP-адрес, то может потребоваться добавить соответствие имени владельца сертификата сервера (CN) этому IP-адресу в hosts-файл используемой ОС:

```
<IP-адрес_установки_туннеля> <имя_владельца_сертификата_(CN)_из_ gateway-  
server.cer>
```

Либо другим способом донести до Stunnel связь:

```
IP-адрес_сервера <--> имя_владельца_сертификата_сервера_CN
```

# Аутентификация

1. Получить у сотрудника банка логин, пароль, секрет и url.
2. Обратиться к серверу авторизации с полученными авторизационными данными:

```
curl --location --request POST
'{authorization_url}/auth/realms/dolinsk/protocol/openid-connect/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id=%идентификатор_клиента%' \
--data-urlencode 'client_secret=%секрет%' \
--data-urlencode 'grant_type=password' \
--data-urlencode 'username=%логин%' \
--data-urlencode 'password=%пароль%'
```

3. В ответе от сервера авторизации будет содержаться токен доступа access\_token.
4. Полученный токен следует добавлять в заголовок Authorization в формате "Bearer " + {access\_token}.

# Процесс межбанковских расчетов

## Описание

Банк-партнер начинает процесс взаимодействия с Банком, передавая исходящий документ. Для этого используется метод **создания исходящего документа** `/mbr/docs/accounts/{accountNumber}` (см. [Схему взаимодействия](#), шаг 1). Успешный ответ от Банка подтверждает, что документ загружен в систему и принят к проверке. При этом метод возвращает идентификатор документа в параметре `doc_id`.

После получения документа Банк проводит его проверку. Содержание документа анализируется, и результат проверки сохраняется в системе. Если документ прошел валидацию и не содержит ошибок, фиксируется статус успешной проверки, и документ принимается к дальнейшей обработке.

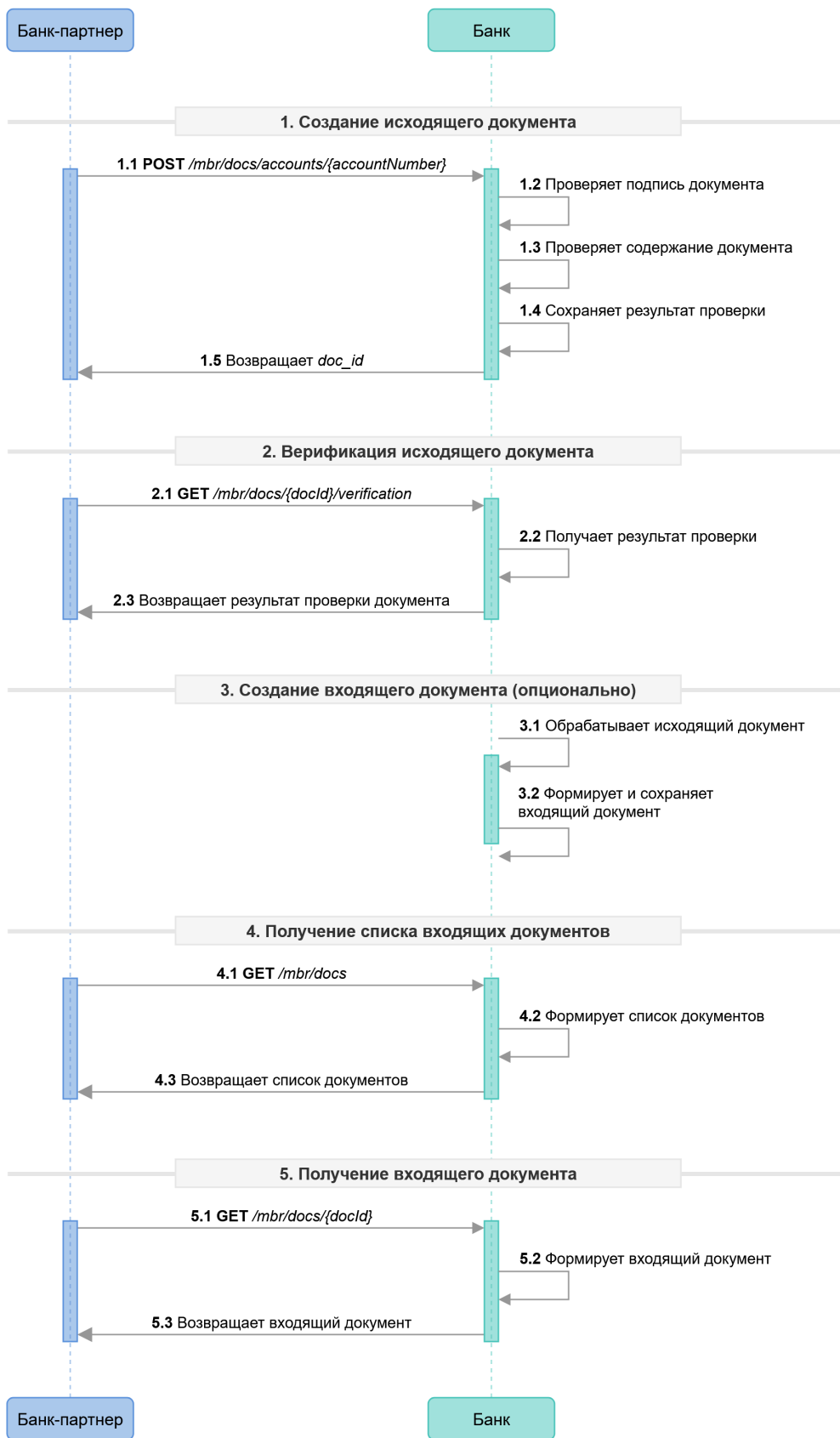
Банк-партнер может **отслеживать статус верификации документа**, отправленного на шаге 1 (см. [Схему взаимодействия](#), шаг 2). Эта проверка осуществляется методом `/mbr/docs/{docId}/verification`, где `docId` – это идентификатор документа, возвращенный Банком на шаге 1. Успешный ответ при проверке статуса содержит одно из четырех значений текущего статуса документа, переданного в параметре `status`. Подробнее о статусах [ниже](#).

Если сценарием взаимодействия Банка и Банка-партнера предусмотрено формирование документов в адрес Банка-партнера, то после формирования их Банком, такие документы будут ожидать запрос на получение от Банка-партнера (см. [Схему взаимодействия](#), шаг 3).

Для получения входящих документов от Банка Банк-партнер выполняет **запрос списка документов**, которые ему еще не были отправлены, методом `/mbr/docs` (см. [Схему взаимодействия](#), шаг 4). В ответ Банк предоставляет список идентификаторов этих документов.

Затем Банк-партнер **запрашивает каждый входящий документ** отдельно методом `/mbr/docs/{docId}` (см. [Схему взаимодействия](#), шаг 5). Здесь `docId` – один из идентификаторов документов, полученных на шаге 4. После успешного получения документа он исключается из списка доступных документов и больше не отображается в ответах на запросы по шагу 4. Срок хранения входящих документов от Банка в системе составляет 30 дней.

# Схема взаимодействия



# Статусы операций, коды ошибок

## Статусы операции проверки верификации исходящего документа

Метод проверки верификации исходящего документа `/mbr/docs/{docId}/verification` может вернуть следующие 4 состояния исходящего документа:

Статус	Описание
START	Начата проверка документа
WAITING	Документ в обработке
SUCCESS	Документ успешно проверен
ERROR	Документ отклонен, произошла ошибка

Если на запрос проверки вернулся статус **ERROR**, то описание проблемы можно получить в параметре `description`, полученном в этом же ответе.

## Справочник кодов детализации ошибок

Код ошибки	Описание ошибки
C001	Ошибка взаимодействия с банком
C002	Элемент не найден
C003	Элемент уже существует
C004	Ошибка валидации входных данных
C005	Метод не поддерживается
C006	Ошибка выполнения операции, подпись недействительна

<b>Код ошибки</b>	<b>Описание ошибки</b>
C007	Ошибка типа сообщения
C025	Недостаточно прав доступа
C027	Пользователь не аутентифицирован
C029	Некорректный счет или данные счета
C042	Срок хранения истек
C043	Размер файла превышает максимально допустимое значение
C043	Размер файла превышает максимально допустимое значение
S000-999	Внутренняя ошибка, попробуйте позже
B000-999	Проверьте правильность передачи данных или обратитесь в техническую поддержку

# Тестовые сценарии

Для завершения интеграции и подключения к промышленному контуру необходимо провести тестирование по всем пунктам, указанным в файле [по этой ссылке](#).

Скачайте этот файл и заполните поля, подкрашенные зеленым цветом, указав соответствующие данные о проведённых тестовых запросах: **номер счета, id операции, дата и время транзакции**. Заполненный файл нужно отправить менеджеру банка с сообщением о завершении интеграции.

## Описание эндпоинтов

<b>Security Scheme Type</b>	http
<b>HTTP Authorization Scheme</b>	bearer
<b>Bearer format</b>	JWT

### Направить документ в Банк

<b>POST</b>	<code>/mbr/docs/accounts/{accountNumber}</code>
-------------	---

Метод создает и направляет документ в Банк.

## Request

### PATH PARAMETERS

**accountNumber** **REQUIRED**

Допустимые значения: Количество символов = 20

Паттерн: `^\d+$`

Пример: `30111810000000000018`

Номер счета

---

### HEADER PARAMETERS

**Signature** **REQUIRED**

string

Подпись тела сообщения

## X-Request-Id

string <uuid>

Идентификатор операции

---

## REQUEST BODY SCHEMA: multipart/form-data

**file** REQUIRED

string <binary>

---

## Responses

200

Результат операции

### RESPONSE SCHEMA: application/json

**doc\_id** REQUIRED

string <uuid>

Пример: 75e48338-f5c1-4c5a-b02b-0a17cf51b58f

Идентификатор зарегистрированного документа

---

400

Некорректный запрос

### RESPONSE SCHEMA: application/json

#### code

string

Пример: E002

Код исключения

#### message

string

Пример: Ошибка жизненного цикла

Сообщение исключения

#### details

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

#### violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**401**

Ошибка аутентификации

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**403**

Ошибка авторизации

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: `fail_url`

Наименование поля с нарушением

**message** REQUIRED

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

**422**

Ошибка при обработке данных

**RESPONSE SCHEMA:** application/json

### code

string

Пример: `E002`

Код исключения

### message

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: `fail_url`

Наименование поля с нарушением

**message** REQUIRED

string

Пример: `должно содержать допустимый URL`

429

Слишком много запросов

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

500

Внутренняя ошибка сервиса

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

## violations

object[]

Информация об ошибках валидации

### field REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

### message REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

## Проверка верификации отправленного документа

GET /mbr/docs/{docId}/verification

Метод возвращает статус верификации исходящего документа. Возможные значения статуса верификации и их значение:

Статус	Описание
START	Начата проверка документа
WAITING	Документ в обработке
SUCCESS	Документ успешно проверен
ERROR	Документ отклонен, произошла ошибка

### Request

#### PATH PARAMETERS

**docId** **REQUIRED**

string <uuid>

Id документа

#### HEADER PARAMETERS

**X-Request-Id**

string <uuid>

Идентификатор операции

### Responses

200

Статус верификации

**RESPONSE SCHEMA:** application/json

**status** **REQUIRED**

string

Пример: START

Статус верификации документа

**description**

string

Пример: Неверная структура документа

Описание причины установки ошибочного статуса при status = ERROR

---

**400**

Некорректный запрос

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**401**

Ошибка аутентификации

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

## details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

## violations

object[]

Информация об ошибках валидации

### field **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

### message **REQUIRED**

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

**403**

Ошибка авторизации

## RESPONSE SCHEMA: application/json

### code

string

Пример: `E002`

Код исключения

### message

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

#### field **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

#### message **REQUIRED**

string

Пример: `должно содержать допустимый URL`

422

Ошибка при обработке данных

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

429

Слишком много запросов

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

**message** **REQUIRED**

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

---

**500**

Внутренняя ошибка сервиса

### RESPONSE SCHEMA: application/json

#### code

string

Пример: `E002`

Код исключения

#### message

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

#### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

**message** **REQUIRED**

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

---

## Получить список документов от Банка

GET

/mbr/docs

Метод возвращает список входящих документов из Банка

### Request

#### QUERY PARAMETERS

##### page

integer <int32>

Допустимые значения: `>= 0`

Значение по умолчанию: `0`

Номер страницы

##### pageSize

integer <int32>

Допустимые значения: `>= 500` `<= 1000`

Значение по умолчанию: `500`

Количество неполученных сообщений на странице

##### dateFrom

string <date>

Начало периода загрузки документов в АБС в формате уууу-мм-дд

##### dateTo

string <date>

Окончание периода загрузки документов в АБС в формате уууу-мм-дд

---

#### HEADER PARAMETERS

##### X-Request-Id

string <uuid>

Идентификатор операции

---

### Responses

200

Список документов

**RESPONSE SCHEMA:** application/json

##### entities **REQUIRED**

string[] <uuid>

##### total\_elements **REQUIRED**

integer <int64>

Пример: `10`

Общее количество элементов

**total\_pages** **REQUIRED**

integer <int32>

Пример: 1

Общее количество страниц

---

**400**

Некорректный запрос

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** **REQUIRED**

string

Пример: fail\_url

Наименование поля с нарушением

**message** **REQUIRED**

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**401**

Ошибка аутентификации

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

#### field **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

#### message **REQUIRED**

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

**403**

Ошибка авторизации

## RESPONSE SCHEMA: application/json

### code

string

Пример: `E002`

Код исключения

### message

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

#### field **REQUIRED**

string

Пример: `fail_url`

Наименование поля с нарушением

#### message **REQUIRED**

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**422**

Ошибка при обработке данных

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**429**

Слишком много запросов

**RESPONSE SCHEMA: application/json**

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: `fail_url`

Наименование поля с нарушением

**message** REQUIRED

string

Пример: `должно содержать допустимый URL`

Сообщение о нарушении валидации

**500**

Внутренняя ошибка сервиса

**RESPONSE SCHEMA:** application/json

### code

string

Пример: `E002`

Код исключения

### message

string

Пример: `Ошибка жизненного цикла`

Сообщение исключения

### details

string

Пример: `Ошибка очередности смены статусов операции`

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: `fail_url`

Наименование поля с нарушением

**message** REQUIRED

string

Пример: `должно содержать допустимый URL`



## Получить входящий документ

GET

/mbr/docs/{docId}

Метод возвращает указанный входящий от Банка документ.

### Request

#### PATH PARAMETERS

**docId** **REQUIRED**

string <uuid>

Id документа

---

#### HEADER PARAMETERS

**X-Request-Id**

string <uuid>

Идентификатор операции

---

### Responses

**200**

Результат операции

**RESPONSE SCHEMA:** application/octet-stream

**body**

string <binary>

---

**400**

Некорректный запрос

**RESPONSE SCHEMA:** application/json

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

## violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**401**

Ошибка аутентификации

**RESPONSE SCHEMA:** application/json

## code

string

Пример: E002

Код исключения

## message

string

Пример: Ошибка жизненного цикла

Сообщение исключения

## details

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

## violations

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**403**

Ошибка авторизации

**RESPONSE SCHEMA:** application/json

**code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---

**422****Ошибка при обработке данных****RESPONSE SCHEMA: application/json****code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

**429**

Слишком много запросов

**RESPONSE SCHEMA: application/json****code**

string

Пример: E002

Код исключения

**message**

string

Пример: Ошибка жизненного цикла

Сообщение исключения

**details**

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

**violations**

object[]

Информация об ошибках валидации

**field** REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

**message** REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

**500**

Внутренняя ошибка сервиса

**RESPONSE SCHEMA: application/json****code**

string

Пример: E002

Код исключения

### message

string

Пример: Ошибка жизненного цикла

Сообщение исключения

### details

string

Пример: Ошибка очередности смены статусов операции

Дополнительные данные об ошибке

### violations

object[]

Информация об ошибках валидации

#### field REQUIRED

string

Пример: fail\_url

Наименование поля с нарушением

#### message REQUIRED

string

Пример: должно содержать допустимый URL

Сообщение о нарушении валидации

---