

Описание API

Содержание

Общая информация и возможности	5
Программный интерфейс	5
Организация заказов и операций	5
Статусы заказов	5
Статусы операций	6
Схемы взаимодействия	6
Оплата через платежную страницу	6
Сервер-сервер	6
Списание средств в бэкофисе	6
Работа через API	7
Аутентификация	7
HTTP-коды	7
Типы отказов	7
Обработка ошибок	7
Обработка отклоненных транзакций	8
Валидация	8
Пример кода обработки	8
Запросы	9
Тестовый запрос (GET /ping)	9
Выборка данных	9
Информация о заказе (GET /orders/:id)	9
Список заказов (GET /orders/)	12
Список операций (GET /operations/)	13
Экспорт курсов валют (GET /exchange_rates/)	14

Работа с заказами	15
Создание заказа (POST /orders/create)	15
Авторизация (POST /orders/authorize)	17
Отмена авторизации (PUT /orders/:id/reverse)	21
Списание средств (PUT /orders/:id/charge)	22
Возврат средств (PUT /orders/:id/refund)	23
Отмена (PUT /orders/:id/cancel)	24
Повторный платеж (POST /orders/:id/rebill)	25
Выплата на карту (OCT, Original credit transaction) (POST /orders/:id/credit)	26
Выплата на карту без ссылки на заказ (OCT, Original credit transaction) (POST /orders/credit)	28
Завершение 3-D Secure аутентификации (POST /orders/:id/complete) ..	29
Завершение 3-D Secure 2.0 аутентификации (POST /orders/:id/complete3d20)	30
Продолжение 3DS Method URL аутентификации (POST /orders/:id/resume)	31
Дополнительные возможности	32
Динамический дескриптор	32
Поддержка длинной записи (Airline Addendum)	33
Уникальная конверсия платежей	33
Платежи с 3-D Secure 2.0	33
DSRP-платежи	34
Привязка платежной карты (Oneclick)	35
Постраничная выгрузка списков ордеров и операций	35
Фискализация по 54-ФЗ	36
Общая информация	36

Общая информация	36
Регистрация чека	36
Запрос реквизитов чека	39
Тестирование	39
Общая информация	39

Общая информация и возможности

Программный интерфейс построен по принципам REST, на основе спецификации JSON-API. API предоставляет единый интерфейс как для интеграции через платежную страницу, так и для работы по схеме сервер-сервер.

Возможности:

- JSON-API на основе стандарта RESTful JSON API.
- Дополнительная аутентификация на основе клиентских сертификатов.
- Одностадийная и двухстадийная схемы оплаты.
- Схемы интеграции: через платежную страницу или сервер-сервер с использованием единого API.
- Поддержка 3-D Secure.
- Повторная оплата без ввода номера, даты истечения и держателя карты.
- Поля для передачи дополнительных данных.
- Выбор сегмента для разделения потока транзакций.
- Выбор языка и шаблона платежной страницы.
- Установка времени жизни сессии.
- Выплаты на карту (кредитные транзакции).
- Передача динамического дескриптора.
- Рекуррентные платежи.
- Мультивалютность.
- Проведение авторизаций с нулевой суммой.
- Проведение DSRP-платежей.
- Поддержка 3-D Secure 2.0.

Программный интерфейс

Организация заказов и операций

Платежи представляются в виде заказов. Заказ хранит общую информацию о платеже и реквизиты держателя. Транзакции представлены в виде последовательности операций, которые ассоциируются с заказом, например: авторизация, финансовое подтверждение, возврат средств.

Созданному заказу присваивается идентификатор, позже этот идентификатор используется в запросах, связанных этим заказом.

Статусы заказов

По статусу заказа можно судить о текущем состоянии.

Статус	Расшифровка	Возможные операции
<code>new</code>	Заказ создан, но обработка еще не началась	—
<code>processing</code>	Выполняется обработка заказа	—
<code>prepared</code>	Ожидаются действия извне	—
<code>authorized</code>	Проведена успешная авторизация	<code>reverse, charge, rebill</code>
<code>charged</code>	Проведено успешное списание средств	<code>refund, rebill</code>
<code>reversed</code>	Проведено успешное аннулирование	<code>rebill</code>

refunded rejected	Проведен успешный возврат средств Заказ был отклонен системой	refund, rebill
fraud	Заказ был определен как фродный и отклонен системой	rebill
declined	Заказ был отклонен банком-эквайером	rebill
chargedback	Заказ был отмечен как чарджбек	rebill
error	В процессе выполнения произошла ошибка, возможно, требуется вмешательство службы поддержки	rebill

Статусы операций

По статусу операции можно судить об успешности ее выполнения.

Статус	Расшифровка
success	Операция проведена успешно
failure	Операция отклонена (например, произошел отказ авторизации)
error	В процессе выполнения произошла ошибка, возможно, требуется вмешательство службы поддержки

Схемы взаимодействия

Оплата через платежную страницу

1. Пользователь нажимает кнопку "Оплатить" на сайте продавца.
2. Продавец обрабатывает запрос пользователя на совершение покупки и отправляет в систему запрос `create`, указывая описание товара, сумму платежа и при необходимости другие данные (см. [Создание заказа](#)).
3. В ответ система передает продавцу `URI` платежной страницы.
4. Продавец перенаправляет пользователя на полученный `URI`.
5. Пользователь вводит платежную информацию в форму.
6. Система обрабатывает форму и перенаправляет пользователя в магазин продавца.
7. Продавец по номеру заказа из `URI` запрашивает состояние платежа у системы (см. [Информация о заказе](#)) и на основе полученного статуса заказа формирует страницу-результат.

Сервер-сервер

1. Пользователь вводит платежную информацию и нажимает кнопку "Оплатить" на сайте продавца.
2. Продавец обрабатывает запрос пользователя на совершение покупки и отправляет в систему запрос `authorize` (см. [Авторизация заказа](#)).
3. Система создает заказ и обрабатывает платеж, используя полученные реквизиты.
4. Система возвращает информацию о результате проведения операции.
5. Продавец обрабатывает результат проведения операции и формирует страницу-результат.

Списание средств в бэкофисе

После успешного проведения оплаты через платежную страницу производится списание средств с карты заказчика. Способы списания:

1. Включение автоклиринга в проекте (задержка автоклиринга задается в минутах).
2. Ручное списание средств в бэкофисе ("Заказы => Искомый заказ => Списать").

Работа через API

- Запросы осуществляются по HTTP 1.1 с использованием SSL и *Basic*-авторизации.
- Кодировка запросов и ответов: *UTF-8*.
- Запросы и ответы в API передаются в формате *JSON*.
- Для выгрузки информации используется метод *GET*, для проведения манипуляций над данными — методы *POST* и *PUT*.
- Для *GET*-запросов параметры запроса передаются как *QUERY STRING*.
- Для *POST*-запросов параметры передаются в теле *POST*-запроса, иногда передается и *QUERY STRING*.
- Успешные ответы отдаются с *HTTP*-статусом 200 (или 201). В случае ошибки статус может отличаться (см. [HTTP-коды](#)).

Аутентификация

Для доступа к API используется *Basic*-авторизация. Для каждого проекта выдается логин и пароль.

```
GET /orders/ HTTP/1.1
Authorization: Basic Zm9vOmJhcg==
Host: example.com
```

HTTP-коды

Дополнительную информацию несет HTTP-код ответа:

Код	Описание
200	Этот код означает, что запрошенная операция была проведена успешно. В зависимости от запрошенной операции это может означать успешно проведенную авторизацию, отмену, списание или возврат.
201	Заказ создан и ожидает ввода данных от пользователя.
402	Операция отклонена эквайером или шлюзом.
422	Ошибка валидации данных запроса.
500	Внутренняя ошибка в эквайере или шлюзе.

Типы отказов

Типы отказов описывают причину отказа:

Код	Описание
<i>declined</i>	Эквайер отклонил транзакцию
<i>fraud</i>	Транзакция отклонена системой защиты от мошеннических транзакций в шлюзе.
<i>rejected</i>	Транзакция отклонена шлюзом.
<i>error</i>	Отказ из-за ошибки на стороне эквайера или шлюза.
<i>validation</i>	Запрос не обработан, т.к. не прошел валидацию.

Обработка ошибок

Все отказы и ошибки возвращаются в едином формате

```
{
  "failure_type" : "error",
  "failure_message" : "This is an error",
```

```
"order_id" : null,
}
```

Причина отказа передается в поле *failure_message*. Если в процессе обработки заказа был создан заказ, его идентификатор передается в поле *order_id*.

Обработка отклоненных транзакций

Все ответы для отклоненных эмитентом транзакций возвращаются в едином формате:

```
{
  "failure_message": "Soft-Decline",
  "order_id": "123456789",
  "failure_type": "declined"
}
```

Принцип формирования ответа такой же, как при обработке [ошибок](#).

Если авторизация отклонена по сценарию Soft-Decline, то это значит, что параметры запроса не удовлетворяют нормативам SCA (Strong Customer Authentication). Такую авторизацию следует проводить с передачей флага *force3d*.

Признак Soft-Decline - в информации о заказе в поле *iso_response_code* будет значение *A1*.

Валидация

В случае ошибки валидации передается дополнительный параметр *errors*, где содержится информация об ошибочных полях. Для удобства передается дополнительная информация о поле, которое вызвало ошибку.

Пример:

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "foo" : "bar"
}' 'http://project:***@api.box:5001/orders/create'

> POST /orders/create HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 13
>
< HTTP/1.1 422 Unprocessable Entity
< Content-Type: application/json
< Content-Length: 223
<
```

```
{ "errors" : [ { "attribute" : "required", "details" : [ "(true)" ], "message" : "Required", "uri" : "#/amount" }, { "message" : "Unknown property", "uri" : "#/foo" } ], "failure_message" : "Validation failed", "failure_type" : "validation", "order_id" : null }
```

Пример кода обработки

```
response = http.post('https://api/authorize', pan => 4111111111111111, ... , cvv => 123);

if response.code == 200 {
  order = decode_json(response.body);
}
```

```

    print "Authorization success";
    print "Order ID: " . order.id

} else if response.code == 402 {

    print "Authorization declined";
    print "Decline reason: " . response.failure_message;

} else if response.code == 422 {

    print "Input data is incorrect";
    print "Reason: " . response.failure_message;

} else {

    print "An error ocured: " . response.failure_message;
}

```

Запросы

Тестовый запрос (GET /ping)

Параметры

Не принимаются.

Ответ

Название	Описание	Пример
message	Сообщение	PONG!
date	Текущие дата и время на сервере	2014-06-30 14:09:20

Пример:

```

curl -v -X GET -H 'Content-Type: application/json' 'http://project:***@api.box:5001/ping'

> GET /ping HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 48
<

```

```
{ "date" : "2016-08-22 15:39:42", "message" : "PONG!" }
```

Выборка данных

При выгрузке данных, таких как, например, список заказов, максимальное количество записей, которые можно получить за один запрос, составляет 2000 записей. Рекомендуется использовать параметры фильтрации, чтобы ограничить количество выгружаемых данных.

Информация о заказе (GET /orders/:id)

Параметры

Name	Description	Example
expand	List of extra objects to display. Possible expands are: <code>card</code> , <code>client</code> , <code>location</code> , <code>custom_fields</code> , <code>issuer</code> , <code>secure3d</code> , <code>operations.cashflow</code>	<code>card,location,client</code>

Ответ

Название	Описание	Пример
<code>id</code>	Уникальный идентификатор заказа	5673044075769513
<code>merchant_order_id</code>	Идентификатор заказа в системе мерчанта	5678
<code>auth_code</code>	Код авторизации	AUTH12
<code>amount</code>	Сумма заказа	9.99
<code>amount_charged</code>	Сумма списанных средств	9.99
<code>amount_refunded</code>	Сумма возвращенных средств	5.99
<code>pan</code>	Номер карты	411111****1111
<code>segment</code>	Сегмент	987654321
<code>status</code>	Статус заказа	authorized
<code>created</code>	Дата создания заказа	2014-06-30 14:38:48
<code>updated</code>	Дата последнего действия над заказом	2014-06-30 14:38:48
<code>card</code>	Информация о карте (*)	—
<code>*.subtype</code>	Подтип карты	classic
<code>*.type</code>	Тип карты	visa
<code>client</code>	Информация о клиенте (*)	—
<code>*.address</code>	Адрес	Main ave. 1
<code>*.city</code>	Город	San Francisco
<code>*.country</code>	Страна	USA
<code>*.email</code>	Е-mail	foo@bar.com
<code>*.name</code>	Имя	Ivanov
<code>*.phone</code>	Телефон	+1 456 890
<code>*.state</code>	Штат	CA
<code>*.zip</code>	Почтовый индекс	123456
<code>custom_fields</code>	Пользовательские поля	—
<code>*.description</code>	Описание заказа	Book sale #453
<code>issuer</code>	Информация о банке-эмитенте	—
<code>*.bin</code>	VIN-код	411111
<code>*.country</code>	Страна	RUS
<code>*.title</code>	Название банка	Bank
<code>location</code>	Местоположение клиента (*)	—
<code>*.ip</code>	IP-адрес	8.8.8.8
<code>operations</code>	Массив выполненных операций над заказом (*)	—

Описание полей см. в разделе [Список операций](#).

Пример:

```
curl -v -X GET -H 'Content-Type: application/json'
'http://project:***@api.box:5001/orders/65213580296542862?
expand=card,client,location,custom_fields,issuer,secure3d,operations.cashflow'

> GET /orders/23014806815093624?
expand=card,client,location,custom_fields,issuer,secure3d,operations.cashflow HTTP/1.1
```

```

> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1018
<
{
  "orders" : [
    {
      "amount" : "9.99",
      "amount_charged" : "0.00",
      "amount_refunded" : "0.00",
      "auth_code" : "AUTH12",
      "card" : {
        "holder" : "John Smith",
        "subtype" : "classic",
        "type" : "visa"
      },
      "client" : {
        "address" : "Main ave. 1",
        "city" : "San Francisco",
        "country" : "USA",
        "email" : "foo@bar.com",
        "name" : "John Smith",
        "phone" : "+1 456 890",
        "state" : "CA",
        "zip" : "123456"
      },
      "created" : "2021-11-12 12:16:13",
      "currency" : "USD",
      "custom_fields" : {},
      "description" : "Book sale #453",
      "descriptor" : "TESTMERCH/TERM",
      "id" : "65213580296542862",
      "issuer" : {
        "bin" : "411111",
        "country" : "UKR",
        "title" : "TestBank"
      },
      "location" : {
        "ip" : "6.6.6.6"
      },
      "merchant_order_id" : "5678",
      "operations" : [
        {
          "type": "authorize",
          "iso_message": "Approved",
          "created": "2021-11-12 12:16:13",
          "amount": "1213.00",
          "auth_code": "212636",
          "iso_response_code": "00",
          "cashflow": {
            "incoming": "0.00",
            "fee": "0.00",
            "receivable": "-36.39",
            "amount": "0.00",
            "currency": "USD",
            "reserve": "36.39"
          },
          "currency": "USD",
          "status": "success"
        },
        {

```

```

    "currency": "USD",
    "status": "success",
    "cashflow": {
      "receivable": "1164.48",
      "amount": "1213.00",
      "reserve": "36.39",
      "currency": "USD",
      "incoming": "1200.87",
      "fee": "12.13"
    },
    "arn": "73922059535879175995901",
    "iso_response_code": "00",
    "iso_message": "Approved",
    "created": "2021-11-12 12:21:13",
    "amount": "1213.00",
    "type": "charge"
  }
],
"pan" : "411111****1111",
"secure3d" : {},
"segment" : "987654321",
"status" : "authorized",
"updated" : "2016-08-22 15:39:49"
}
]
}

```

Список заказов (GET /orders/)

Параметры

Name	Description	Example
<code>expand</code>	Список расширений, которые требуется показать. Возможные расширения: <code>card</code> , <code>client</code> , <code>location</code> , <code>custom_fields</code> , <code>issuer</code> , <code>secure3d</code> , <code>operations.cashflow</code>	<code>card,location,client</code>
<code>status</code>	Фильтр по статусу	<code>charged,refunded</code>
<code>created_from</code>	Фильтр по дате создания	<code>2016-01-01 00:00:00</code>
<code>created_to</code>	Фильтр по дате создания	<code>2016-01-31 23:59:59</code>
<code>merchant_order_id</code>	Фильтр по идентификатору	<code>12345,123456</code>
<code>card.type</code>	Фильтр по типу карты	<code>visa,mastercard</code>
<code>card.subtype</code>	Фильтр по субтипу карты	<code>electron,maestro</code>
<code>location.ip</code>	Фильтр по IP-адресу	<code>8.8.8.8,8.8.8.9</code>
<code>client.name</code>	Поиск по имени клиента или по части имени	<code>Smith</code>
<code>client.email</code>	Поиск по адресу электронной почты или по части адреса	<code>@yahoo.com</code>
<code>issuer.country</code>	Фильтр по стране эмитента (ISO 3166-1 alpha-3)	<code>RUS</code>
<code>issuer.bin</code>	Фильтр по БИН-коду	<code>555555,411111</code>
<code>issuer.title</code>	Фильтр по названию эмитента	<code>Credit Union</code>

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -v -X GET -H 'Content-Type: application/json' 'http://project:***@api.box:5001/orders/'

> GET /orders/ HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 5550
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "auth_code" : "AUTH12",
"created" : "2016-08-22 15:39:49", "currency" : "USD", "description" : "Book sale #453", "descriptor" :
"TESTMERCH/TERM", "id" : "23014806815093624", "merchant_order_id" : "5678", "pan" : "411111****1111",
"segment" : "987654321", "status" : "authorized", "updated" : "2016-08-22 15:39:49" }, { "amount" : "9.99",
"amount_charged" : "0.00", "amount_refunded" : "0.00", "auth_code" : "AUTH12", "created" : "2016-08-22 15:39:48",
"currency" : "USD", "description" : null, "descriptor" : "TESTMERCH/TERM", "id" : "23014806643308174",
"merchant_order_id" : null, "pan" : "411111****1111", "status" : "authorized", "updated" : "2016-08-22 15:39:48" } ] }
```

Список операций (GET /operations/)

Параметры

Name	Description	Example
<code>expand</code>	Список расширений которые требуется показать. Возможные расширения: <code>cashflow, extra</code>	<code>cashflow</code>
<code>status</code>	Фильтр по статус	<code>error</code>
<code>type</code>	Фильтр по типу	<code>charge, refund</code>
<code>created_from</code>	Фильтр по дате создания	<code>2016-01-01 00:00:00</code>
<code>created_to</code>	Фильтр по дате создания	<code>2016-01-31 23:59:59</code>

Ответ

Название	Описание	Пример
<code>amount</code>	Сумма операции	<code>9.99</code>
<code>auth_code</code>	Код авторизации	<code>AUTH12</code>
<code>created</code>	Дата проведения операции	<code>2014-06-30 14:38:48</code>
<code>iso_message</code>	Сообщение	<code>Approved</code>
<code>iso_response_code</code>	Код ответа	<code>00</code>
<code>status</code>	Статус операции	<code>success</code>
<code>type</code>	Тип операции	<code>authorize</code>
<code>cashflow</code>	Финансовая информация	—
<code>*.amount</code>	Сумма	<code>9.99</code>
<code>*.fee</code>	Комиссия	<code>0.00</code>
<code>*.incoming</code>	Доход	<code>6.00</code>
<code>*.receivable</code>	К получению	<code>5.00</code>

Пример:

```
curl -v -X GET -H 'Content-Type: application/json' 'http://project:***@api.box:5001/operations/'

> GET /operations/ HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
```

```

> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 2931
<
{
  "operations" : [
    {
      "amount" : "11991.00",
      "auth_code" : "AUTH12",
      "created" : "2016-08-24 12:07:00",
      "currency" : "EUR",
      "iso_message" : "Approved",
      "iso_response_code" : "00",
      "order_id" : "23011537206894183",
      "arn": "28281615742473953725862",
      "status" : "success",
      "type" : "charge"
    },
    {
      "amount" : "8997.00",
      "auth_code" : "AUTH12",
      "created" : "2016-08-23 12:06:55",
      "currency" : "USD",
      "iso_message" : "Approved",
      "iso_response_code" : "00",
      "order_id" : "23011536237676397",
      "status" : "success",
      "type" : "authorize"
    }
  ]
}

```

Экспорт курсов валют (GET /exchange_rates/)

Обязательные_параметры

Название	Описание	Пример
<code>date</code>	Дата курсов валют	2014-06-30
<code>date_from</code>	Начальная дата для курсов валют	2021-06-29
<code>date_to</code>	Конечная дата для курсов валют	2021-06-30

Ответ

Название	Описание	Пример
<code>exchange_rates</code>	Обменные данные	—
<code>*.from</code>	Котируемая валюта	USD
<code>*.to</code>	Базовая валюта	EUR
<code>*.rate</code>	Котировка	0.800000

Пример:

```

curl -v -X GET -H 'Content-Type: application/json'
'http://project:***@api.box:5001/exchange_rates/?date=2014-06-30'

> GET /exchange_rates/ HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001

```

```
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1805
<
```

```
{ "exchange_rates" : [ { "from" : "EUR", "rate" : "75.132109", "to" : "RUB" }, { "to" : "UAH", "from" : "EUR", "rate" : "31.562887" }, { "from" : "EUR", "rate" : "334.000000", "to" : "UBT" }, { "to" : "USD", "from" : "EUR", "rate" : "1.145036" } ] }
```

Примечание

Не допускается одновременное использование фильтров по дате и диапазону дат.

Работа с заказами

Создание заказа (POST /orders/create)

Успешный ответ на этот запрос возвращается с кодом 201. Это означает, что заказ создан и он ожидает ввода данных от держателя карты.

Обязательные параметры

Название	Описание	Пример
<code>amount</code>	Сумма заказа	9.99

Необязательные параметры

Название	Описание	Пример
<code>currency</code>	Order currency	USD
<code>merchant_order_id</code>	Идентификатор заказа в системе мерчанта	5678
<code>segment</code>	Сегмент	987654321
<code>description</code>	Описание заказа	Book sale #453
<code>client</code>	Информация о клиенте (*)	—
<code>*.address</code>	Адрес	Main ave. 1
<code>*.city</code>	Город	San Francisco
<code>*.country</code>	Страна	USA
<code>*.login</code>	Логин клиента	john_doe
<code>*.email</code>	E-mail	foo@bar.com
<code>*.name</code>	Имя	Ivanov
<code>*.phone</code>	Телефон	+1 456 890
<code>*.state</code>	Штат	CA
<code>*.zip</code>	Почтовый индекс	123456
<code>custom_fields</code>	Пользовательские поля (*)	—
<code>location</code>	Местоположение клиента (*)	—
<code>*.ip</code>	IP-адрес	8.8.8.8
<code>options</code>	Опции (*)	—
<code>*.expiration_timeout</code>	Время сессии	30m
<code>*.force3d</code>	Принудительное направление в 3-D Secure	1
<code>*.auto_charge</code>	Провести операцию Charge (используется для одностадийных платежей)	1

*.language	Язык платежной формы	ru
*.return_url	URL для перенаправления клиента после оплаты	http://my.url
*.template	Шаблон платежной формы	12
*.mobile	Мобильный шаблон	1
*.terminal	Терминал	TERM12
*.apple_pay_enabled	Включение поддержки Apple Pay на платежной странице	1
*.google_pay_enabled	Включение поддержки Google Pay™ на платежной странице	1
*.secure3d20_return_url	URL для уведомлений 3-D Secure 2.0	http://return.url
*.exemption_mit	Флаг Exemption MIT	1

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Дополнительно возвращается заголовок *Location* в котором указывается ссылка на платежную страницу.

Пример (минимальный набор данных):

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99
}' 'http://project:***@api.box:5001/orders/create'

> POST /orders/create HTTP/1.1
> Authorization: Basic CHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 15
>
< Location: http://localhost:5002/pay/23014805213622230
< Content-Type: application/json
< Content-Length: 341
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "card" : {}, "client" : {},
"created" : "2016-08-22 15:39:42", "currency" : "USD", "custom_fields" : {}, "description" : null, "id" :
"23014805213622230", "issuer" : {}, "location" : {}, "merchant_order_id" : null, "operations" : [], "secure3d" : {},
"status" : "new", "updated" : "2016-08-22 15:39:42" } ] }
```

Пример (расширенный набор данных):

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "client" : {
    "address" : "Main ave. 1",
    "city" : "San Francisco",
    "country" : "USA",
    "email" : "foo@bar.com",
    "name" : "John Smith",
    "phone" : "+1 456 890",
    "state" : "CA",
    "zip" : "123456"
  },
  "description" : "Book sale #453",
  "location" : {
    "ip" : "6.6.6.6"
  },
  "merchant_order_id" : "5678",
  "options" : {
```

```

    "force3d" : 0,
    "return_url" : "http://shop.com",
    "terminal" : ""
  },
  "segment" : "987654321"
}' 'http://project:***@api.box:5001/orders/create'

> POST /orders/create HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 357
>
< HTTP/1.1 201 Created
< Location: http://localhost:5002/pay/23014805439541518
< Content-Type: application/json
< Content-Length: 544
<

```

```

{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "card" : {}, "client" : {
"address" : "Main ave. 1", "city" : "San Francisco", "country" : "USA", "email" : "foo@bar.com", "name" : "John
Smith", "phone" : "+1 456 890", "state" : "CA", "zip" : "123456" }, "created" : "2016-08-22 15:39:43", "currency" :
"USD", "custom_fields" : {}, "description" : "Book sale #453", "id" : "23014805439541518", "issuer" : {}, "location" : {
"ip" : "6.6.6.6" }, "merchant_order_id" : "5678", "operations" : [], "secure3d" : {}, "segment" : "987654321", "status" :
"new", "updated" : "2016-08-22 15:39:43" } ] }

```

Примечание

Для управления включением *Google Pay*[™] и *Apple Pay* на платежной странице с помощью флагов `*_pay_enabled` глобальная поддержка должна быть отключена в настройках.

Авторизация (POST /orders/authorize)

Обязательные параметры

Название	Описание	Пример
<code>amount</code>	Сумма заказа	9.99
<code>pan</code>	Номер карты	4111111111111111
<code>card</code>	Информация о карте	—
<code>*.cvv</code>	CVV-код (может быть отключен в настройках проекта)	123
<code>*.holder</code>	Держатель карты (от 2 до 40 символов, может быть отключен в настройках проекта)	John Smith
<code>*.expiration_month</code>	Месяц истечения карты	06
<code>*.expiration_year</code>	Год истечения карты	2022
<code>location</code>	Местоположение клиента (*)	—
<code>*.ip</code>	IP-адрес	8.8.8.8

Необязательные параметры

Название	Описание	Пример
<code>currency</code>	Валюта	USD
<code>merchant_order_id</code>	Идентификатор заказа в системе мерчанта	5678
<code>segment</code>	Сегмент	987654321
<code>description</code>	Описание заказа	Book sale #453
<code>client</code>	Информация о клиенте (*)	—
<code>*.address</code>		Main ave. 1

*.city	Адрес	San Francisco
*.country	Страна	USA
*.email	E-mail	foo@bar.com
*.name	Имя	Ivanov
*.phone	Телефон	+1 456 890
*.state	Штат	CA
*.zip	Почтовый индекс	123456
custom_fields	Пользовательские поля (*)	—
location	Местоположение клиента (*)	—
*.ip	IP-адрес	8.8.8.8
options	Опции (*)	—
*.force3d	Принудительное направление в 3-D Secure	1
*.return_url	URL для перенаправления клиента после оплаты	http://my.url
*.auto_charge	Провести операцию Charge (используется для одностадийных платежей)	1
*.terminal	Терминал	TERM12
*.recurring	Флаг рекуррентного платежа	1
*.secure3d20_return_url	URL для уведомлений 3-D Secure 2.0	http://return.url
*.exemption_mit	Флаг Exemption MIT	1
secure3d	Данные аутентификации 3-D Secure (*)	—
*.xid	Идентификатор транзакции из PaRes (XID)	8Rg2xYbxLUGsfZ1koh1JxAQFBM=
*.cavv	Значение CAVV из PaRes	AAABCBVRA2EYZXQBV1EDAAAAAA=
*.eci	Признак ECI	05
*.ds_trans_id	Идентификатор транзакции на DS	Rcm70YAaT/7SZnqVGFQkm/NJnbQ=

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#) и дополнительно могут содержать:

Название	Описание	Пример
form3d_html	3-D Secure форма в формате HTML	<form id="form3d" ...
form3d	3-D Secure форма (*)	—
*.MD	Параметр MD	NTY3NDg0NjMzMjA0MDU1OQ==
*.PaReq	Параметр PaReq	Y3NDg0NjMzM0MDU1OQ
*.TermUrl	URL системы для завершения операции	http://box.com/complete
*.action	URL ACS банка-эквайера	http://bank.com
*.method	Метод формы	POST

Пример (минимальный набор данных):

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "card" : {
    "cvv" : "333",
    "expiration_month" : 12,
    "expiration_year" : 2020,
    "holder" : "John Smith"
  },
  "location" : {
    "ip" : "6.6.6.6"
  },
}
```

```
"pan" : "4111111111111111"
}' 'http://project:***@api.box:5001/orders/authorize'

> POST /orders/authorize HTTP/1.1
> Authorization: Basic CHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 156
>
< Content-Type: application/json
< Content-Length: 829
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "auth_code" : "AUTH12",
"card" : { "holder" : "John Smith", "subtype" : "classic", "type" : "visa" }, "client" : {}, "created" : "2016-08-22 15:39:48",
"currency" : "USD", "custom_fields" : {}, "description" : null, "descriptor" : "TESTMERCH/TERM", "id" :
"23014806643308174", "issuer" : { "bin" : "411111", "country" : "UKR", "title" : "TestBank" }, "location" : { "ip" :
"6.6.6.6" }, "merchant_order_id" : null, "operations" : [ { "amount" : "9.99", "auth_code" : "AUTH12", "cashflow" : {
"amount" : "0.00", "currency" : "USD", "fee" : "0.00", "incoming" : "0.00", "receivable" : "0.00", "reserve" : "0.00" },
"created" : "2016-08-22 15:39:48", "currency" : "USD", "iso_message" : "Approved", "iso_response_code" : "00",
"status" : "success", "type" : "authorize" } ], "pan" : "411111****1111", "secure3d" : {}, "status" : "authorized",
"updated" : "2016-08-22 15:39:48" } ] }
```

Пример (расширенный набор данных):

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "card" : {
    "cvv" : "333",
    "expiration_month" : 12,
    "expiration_year" : 2020,
    "holder" : "John Smith"
  },
  "client" : {
    "address" : "Main ave. 1",
    "city" : "San Francisco",
    "country" : "USA",
    "email" : "foo@bar.com",
    "name" : "John Smith",
    "phone" : "+1 456 890",
    "state" : "CA",
    "zip" : "123456"
  },
  "description" : "Book sale #453",
  "location" : {
    "ip" : "6.6.6.6"
  },
  "merchant_order_id" : "5678",
  "options" : {
    "force3d" : 0,
    "return_url" : "http://shop.com",
    "terminal" : ""
  },
  "pan" : "4111111111111111",
  "segment" : "987654321"
}' 'http://project:***@api.box:5001/orders/authorize'

> POST /orders/authorize HTTP/1.1
> Authorization: Basic CHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 470
```

```
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1018
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "auth_code" : "AUTH12",
"card" : { "holder" : "John Smith", "subtype" : "classic", "type" : "visa" }, "client" : { "address" : "Main ave. 1", "city" :
"San Francisco", "country" : "USA", "email" : "foo@bar.com", "name" : "John Smith", "phone" : "+1 456 890", "state" :
"CA", "zip" : "123456" }, "created" : "2016-08-22 15:39:49", "currency" : "USD", "custom_fields" : {}, "description" :
"Book sale #453", "descriptor" : "TESTMERCH/TERM", "id" : "23014806815093624", "issuer" : { "bin" : "411111",
"country" : "UKR", "title" : "TestBank" }, "location" : { "ip" : "6.6.6.6" }, "merchant_order_id" : "5678", "operations" : [ {
"amount" : "9.99", "auth_code" : "AUTH12", "cashflow" : { "amount" : "0.00", "currency" : "USD", "fee" : "0.00",
"incoming" : "0.00", "receivable" : "0.00", "reserve" : "0.00" }, "created" : "2016-08-22 15:39:49", "currency" : "USD",
"iso_message" : "Approved", "iso_response_code" : "00", "status" : "success", "type" : "authorize" } ], "pan" :
"411111****1111", "secure3d" : {}, "segment" : "987654321", "status" : "authorized", "updated" : "2016-08-22
15:39:49" } ] }
```

Пример (3-D Secure):

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "card" : {
    "cvv" : "333",
    "expiration_month" : 12,
    "expiration_year" : 2020,
    "holder" : "John Smith"
  },
  "location" : {
    "ip" : "6.6.6.6"
  },
  "options" : {
    "force3d" : 1
  },
  "pan" : "4111111111111111"
}' 'http://project:***@api.box:5001/orders/authorize'
```

```
> POST /orders/authorize HTTP/1.1
> Authorization: Basic CHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 180
>
< HTTP/1.1 201 Created
< Content-Type: application/json
< Content-Length: 1083
<
{
  "orders" : [
    {
      "amount" : "9.99",
      "amount_charged" : "0.00",
      "amount_refunded" : "0.00",
      "card" : {
        "holder" : "John Smith",
        "subtype" : "classic",
        "type" : "visa"
      },
      "client" : {},
      "created" : "2016-08-22 15:39:46",
      "currency" : "USD",
```

```

    "custom_fields" : {},
    "description" : null,
    "form3d" : {
      "MD" : "MjMwMTQ4MDYwNzgwNjEzMMDM=",
      "PaReq" : "PaReq",
      "TermUrl" : "http://localhost:5002/complete/secure3d",
      "action" : "http://localhost:5002/test/bank",
      "method" : "GET"
    },
    "form3d_html" : "<form id=\"form3d\" method=\"GET\"
action=\"http://localhost:5002/test/bank\"><input type=\"hidden\" name=\"MD\"
value=\"MjMwMTQ4MDYwNzgwNjEzMMDM=\"><input type=\"hidden\" name=\"PaReq\" value=\"PaReq\"><input
type=\"hidden\" name=\"TermUrl\" value=\"http://localhost:5002/complete/secure3d\"></form>
<script>>window.onload = function() {document.forms['form3d'].submit();}</script>",
    "id" : "23014806078061303",
    "issuer" : {
      "bin" : "411111",
      "country" : "UKR",
      "title" : "TestBank"
    },
    "location" : {
      "ip" : "6.6.6.6"
    },
    "merchant_order_id" : null,
    "operations" : [],
    "pan" : "411111****1111",
    "secure3d" : {
      "reason" : "force3d",
      "scenario" : "unknown"
    },
    "status" : "prepared",
    "updated" : "2016-08-22 15:39:46"
  }
]
}

```

Примечание

После успешной авторизации необходимо провести операцию [Списание средств](#), которая проводит списание средств с карты. Если в настройках аккаунта не установлено автоматическое списание, и в запросе не была передана опция *auto_charge*, и операция *charge* не была проведена явно - списание проведено не будет, а сумма авторизации через некоторое время разблокируются эмитентом и средства вернуться держателю карты.

Отмена авторизации (PUT /orders/:id/reverse)

Параметры

Не принимаются.

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```

curl -v -X PUT -H 'Content-Type: application/json'
'http://project:***@api.box:5001/orders/23014805692462630/reverse'

> PUT /orders/23014805692462630/reverse HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==

```

```
> Host: api.box:5001
> Content-Type: application/json
>
< Content-Type: application/json
< Content-Length: 1115
<
```

```
{ "orders": [ { "amount": "9.99", "amount_charged": "0.00", "amount_refunded": "0.00", "auth_code": "AUTH12",
"card": { "holder": "John Smith", "subtype": "classic", "type": "visa" }, "client": {}, "created": "2016-08-22 15:39:44",
"currency": "USD", "custom_fields": {}, "description": null, "descriptor": "TESTMERCH/TERM", "id":
"23014805692462630", "issuer": { "bin": "411111", "country": "UKR", "title": "TestBank" }, "location": { "ip":
"6.6.6.6" }, "merchant_order_id": null, "operations": [ { "amount": "9.99", "auth_code": "AUTH12", "cashflow": {
"amount": "0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" },
"created": "2016-08-22 15:39:45", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00",
"status": "success", "type": "authorize" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount":
"0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" }, "created":
"2016-08-22 15:39:50", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status":
"success", "type": "reverse" } ], "pan": "411111****1111", "secure3d": {}, "status": "reversed", "updated": "2016-
08-22 15:39:50" } ] }
```

Списание средств (PUT /orders/:id/charge)

Параметры

Название	Описание	Пример
amount	Сумма списываемых средств	9.99

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример (полное списание):

```
curl -v -X PUT -H 'Content-Type: application/json'
'http://project:***@api.box:5001/orders/23014806643308174/charge'

> PUT /orders/23014806643308174/charge HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1113
<
```

```
{ "orders": [ { "amount": "9.99", "amount_charged": "9.99", "amount_refunded": "0.00", "auth_code": "AUTH12",
"card": { "holder": "John Smith", "subtype": "classic", "type": "visa" }, "client": {}, "created": "2016-08-22 15:39:48",
"currency": "USD", "custom_fields": {}, "description": null, "descriptor": "TESTMERCH/TERM", "id":
"23014806643308174", "issuer": { "bin": "411111", "country": "UKR", "title": "TestBank" }, "location": { "ip":
"6.6.6.6" }, "merchant_order_id": null, "operations": [ { "amount": "9.99", "auth_code": "AUTH12", "cashflow": {
"amount": "0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" },
"created": "2016-08-22 15:39:48", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00",
"status": "success", "type": "authorize" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount":
"9.99", "currency": "USD", "fee": "0.30", "incoming": "9.69", "receivable": "9.69", "reserve": "0.00" }, "created":
"2016-08-22 15:39:53", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status":
"success", "type": "charge" } ], "pan": "411111****1111", "secure3d": {}, "status": "charged", "updated": "2016-08-
```

22 15:39:53"]] }

Пример (частичное списание):

```
curl -v -X PUT -H 'Content-Type: application/json' -d '{
  "amount" : 1.99
}' 'http://project:***@api.box:5001/orders/23014806312026412/charge'

> PUT /orders/23014806312026412/charge HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 15
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1113
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "1.99", "amount_refunded" : "0.00", "auth_code" : "AUTH12",
"card" : { "holder" : "John Smith", "subtype" : "classic", "type" : "visa" }, "client" : {}, "created" : "2016-08-22 15:39:47",
"currency" : "USD", "custom_fields" : {}, "description" : null, "descriptor" : "TESTMERCH/TERM", "id" :
"23014806312026412", "issuer" : { "bin" : "411111", "country" : "UKR", "title" : "TestBank" }, "location" : { "ip" :
"6.6.6.6" }, "merchant_order_id" : null, "operations" : [ { "amount" : "9.99", "auth_code" : "AUTH12", "cashflow" : {
"amount" : "0.00", "currency" : "USD", "fee" : "0.00", "incoming" : "0.00", "receivable" : "0.00", "reserve" : "0.00" },
"created" : "2016-08-22 15:39:47", "currency" : "USD", "iso_message" : "Approved", "iso_response_code" : "00",
"status" : "success", "type" : "authorize" }, { "amount" : "1.99", "auth_code" : "AUTH12", "cashflow" : { "amount" :
"1.99", "currency" : "USD", "fee" : "0.06", "incoming" : "1.93", "receivable" : "1.93", "reserve" : "0.00" }, "created" :
"2016-08-22 15:39:51", "currency" : "USD", "iso_message" : "Approved", "iso_response_code" : "00", "status" :
"success", "type" : "charge" } ], "pan" : "411111****1111", "secure3d" : {}, "status" : "charged", "updated" : "2016-08-
22 15:39:51" } ] }
```

Возврат средств (PUT /orders/:id/refund)

Параметры

Название	Описание	Пример
amount	Сумма возвращаемых средств	9.99

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

Пример (полный возврат):

```
curl -v -X PUT -H 'Content-Type: application/json'
'http://project:***@api.box:5001/orders/23014806643308174/refund'

> PUT /orders/23014806643308174/refund HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< Content-Type: application/json
< Content-Length: 1404
<
```

```
{ "orders": [ { "amount": "9.99", "amount_charged": "9.99", "amount_refunded": "9.99", "auth_code": "AUTH12",
"card": { "holder": "John Smith", "subtype": "classic", "type": "visa" }, "client": {}, "created": "2016-08-22 15:39:48",
"currency": "USD", "custom_fields": {}, "description": null, "descriptor": "TESTMERCH/TERM", "id":
"23014806643308174", "issuer": { "bin": "411111", "country": "UKR", "title": "TestBank" }, "location": { "ip":
"6.6.6.6" }, "merchant_order_id": null, "operations": [ { "amount": "9.99", "auth_code": "AUTH12", "cashflow": {
"amount": "0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" },
"created": "2016-08-22 15:39:48", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00",
"status": "success", "type": "authorize" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount":
"9.99", "currency": "USD", "fee": "0.30", "incoming": "9.69", "receivable": "9.69", "reserve": "0.00" }, "created":
"2016-08-22 15:39:53", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status":
"success", "type": "charge" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount": "-9.99",
"currency": "USD", "fee": "0.00", "incoming": "-9.99", "receivable": "-9.99", "reserve": "0.00" }, "created": "2016-
08-22 15:39:55", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status": "success",
"type": "refund" } ], "pan": "411111****1111", "secure3d": {}, "status": "refunded", "updated": "2016-08-22
15:39:55" } ] }
```

Пример (частичный возврат):

```
curl -v -X PUT -H 'Content-Type: application/json' -d '{
  "amount": 1.99
}' 'http://project:***@api.box:5001/orders/23014806312026412/refund'

> PUT /orders/23014806312026412/refund HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 15
>
< Content-Type: application/json
< Content-Length: 1404
<
```

```
{ "orders": [ { "amount": "9.99", "amount_charged": "1.99", "amount_refunded": "1.99", "auth_code": "AUTH12",
"card": { "holder": "John Smith", "subtype": "classic", "type": "visa" }, "client": {}, "created": "2016-08-22 15:39:47",
"currency": "USD", "custom_fields": {}, "description": null, "descriptor": "TESTMERCH/TERM", "id":
"23014806312026412", "issuer": { "bin": "411111", "country": "UKR", "title": "TestBank" }, "location": { "ip":
"6.6.6.6" }, "merchant_order_id": null, "operations": [ { "amount": "9.99", "auth_code": "AUTH12", "cashflow": {
"amount": "0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" },
"created": "2016-08-22 15:39:47", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00",
"status": "success", "type": "authorize" }, { "amount": "1.99", "auth_code": "AUTH12", "cashflow": { "amount":
"1.99", "currency": "USD", "fee": "0.06", "incoming": "1.93", "receivable": "1.93", "reserve": "0.00" }, "created":
"2016-08-22 15:39:51", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status":
"success", "type": "charge" }, { "amount": "1.99", "auth_code": "AUTH12", "cashflow": { "amount": "-1.99",
"currency": "USD", "fee": "0.00", "incoming": "-1.99", "receivable": "-1.99", "reserve": "0.00" }, "created": "2016-
08-22 15:39:52", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status": "success",
"type": "refund" } ], "pan": "411111****1111", "secure3d": {}, "status": "refunded", "updated": "2016-08-22
15:39:52" } ] }
```

Отмена (PUT /orders/:id/cancel)

Аналогично проведению [Отмены авторизации](#) или [Возврата средств](#).

Параметры

Название	Описание	Пример

amount	Сумма возвращаемых средств (кроме аннулирования)	9.99
--------	--	------

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -v -X PUT -H 'Content-Type: application/json'
'http://project:***@api.box:5001/orders/23014808614292147/cancel'

> PUT /orders/23014808614292147/cancel HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 1404
<
```

```
{ "orders": [ { "amount": "9.99", "amount_charged": "9.99", "amount_refunded": "9.99", "auth_code": "AUTH12",
"card": { "holder": "John Smith", "subtype": "classic", "type": "visa" }, "client": {}, "created": "2016-08-22 15:39:56",
"currency": "USD", "custom_fields": {}, "description": null, "descriptor": "TESTMERCH/TERM", "id":
"23014808614292147", "issuer": { "bin": "411111", "country": "UKR", "title": "TestBank" }, "location": { "ip":
"6.6.6.6" }, "merchant_order_id": null, "operations": [ { "amount": "9.99", "auth_code": "AUTH12", "cashflow": {
"amount": "0.00", "currency": "USD", "fee": "0.00", "incoming": "0.00", "receivable": "0.00", "reserve": "0.00" },
"created": "2016-08-22 15:39:56", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00",
"status": "success", "type": "authorize" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount":
"9.99", "currency": "USD", "fee": "0.30", "incoming": "9.69", "receivable": "9.69", "reserve": "0.00" }, "created":
"2016-08-22 15:39:56", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status":
"success", "type": "charge" }, { "amount": "9.99", "auth_code": "AUTH12", "cashflow": { "amount": "-9.99",
"currency": "USD", "fee": "0.00", "incoming": "-9.99", "receivable": "-9.99", "reserve": "0.00" }, "created": "2016-
08-22 15:39:57", "currency": "USD", "iso_message": "Approved", "iso_response_code": "00", "status": "success",
"type": "refund" } ], "pan": "411111****1111", "secure3d": {}, "status": "refunded", "updated": "2016-08-22
15:39:57" } ] }
```

Повторный платеж (POST /orders/:id/rebill)

Обязательные параметры

Название	Описание	Пример
amount	Сумма	9.99
cvv	CVV-код (может быть отключен в настройках проекта)	123
location	Местоположение клиента (*)	—
*.ip	IP-адрес	8.8.8.8

Необязательные параметры

Название	Описание	Пример
options	Опции (*)	—
*.recurring	Флаг рекуррентного платежа	1

Остальные необязательные параметры те же, что и в [Авторизации](#).

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "cvv" : "123"
}' 'http://project:***@api.box:5001/orders/23014808614292147/rebill'

> POST /orders/23014808614292147/rebill HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 27
>
< Content-Type: application/json
< Content-Length: 815
<
```

```
{ "orders" : [ { "amount" : "9.99", "amount_charged" : "0.00", "amount_refunded" : "0.00", "auth_code" : "AUTH12",
"card" : { "holder" : "John Smith", "subtype" : "classic", "type" : "visa" }, "client" : {}, "created" : "2016-08-22 15:39:58",
"currency" : "USD", "custom_fields" : {}, "description" : null, "descriptor" : "TESTMERCH/TERM", "id" :
"23014809139351770", "issuer" : { "bin" : "411111", "country" : "UKR", "title" : "TestBank" }, "location" : {},
"merchant_order_id" : null, "operations" : [ { "amount" : "9.99", "auth_code" : "AUTH12", "cashflow" : { "amount" :
"0.00", "currency" : "USD", "fee" : "0.00", "incoming" : "0.00", "receivable" : "0.00", "reserve" : "0.00" }, "created" :
"2016-08-22 15:39:58", "currency" : "USD", "iso_message" : "Approved", "iso_response_code" : "00", "status" :
"success", "type" : "authorize" } ], "pan" : "411111****1111", "secure3d" : {}, "status" : "authorized", "updated" :
"2016-08-22 15:39:58" } ] }
```

Выплата на карту (ОСТ, Original credit transaction) (POST /orders/:id/credit)

Обязательные параметры

Название	Описание	Пример
amount	Сумма	9.99

Необязательные параметры

Необязательные параметры те же, что и в [Авторизации](#).

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -v -X POST -H 'Content-Type: application/json' -d '{
  "amount" : 9.99,
  "currency" : "USD"
  "location" : {
    "ip" : "8.8.8.8"
  }
}' 'http://project:***@api.box:5001/orders/23014808614292147/credit'

> POST /orders/23014808614292147/credit HTTP/1.1
> Authorization: Basic cHJvamVjdDpwYXNzd29yZA==
```

```

> Host: api.box:5001
> Content-Type: application/json
> Content-Length: 27
>
< Content-Type: application/json
< Content-Length: 815
<
{
  "orders" : [
    {
      "amount" : "9.99",
      "amount_charged" : "0.00",
      "amount_refunded" : "0.00",
      "auth_code" : "AUTH12",
      "card" : {
        "holder" : "John Smith",
        "subtype" : "classic",
        "type" : "visa"
      },
      "client" : {},
      "created" : "2016-08-22 15:39:58",
      "currency" : "USD",
      "custom_fields" : {},
      "description" : null,
      "descriptor" : "TESTMERCH/TERM",
      "id" : "23014809139351770",
      "issuer" : {
        "bin" : "411111",
        "country" : "UKR",
        "title" : "TestBank"
      },
      "location" : {},
      "merchant_order_id" : null,
      "operations" : [
        {
          "amount" : "9.99",
          "auth_code" : "AUTH12",
          "cashflow" : {
            "amount" : "0.00",
            "currency" : "USD",
            "fee" : "0.00",
            "incoming" : "0.00",
            "receivable" : "0.00",
            "reserve" : "0.00"
          },
          "created" : "2016-08-22 15:39:58",
          "currency" : "USD",
          "iso_message" : "Approved",
          "iso_response_code" : "00",
          "status" : "success",
          "type" : "credit"
        }
      ],
      "pan" : "411111****1111",
      "secure3d" : {},
      "status" : "credited",
      "updated" : "2016-08-22 15:39:58"
    }
  ]
}

```

Выплата на карту без ссылки на заказ (OCT, Original credit transaction) (POST)

/orders/credit)

Обязательные параметры

Название	Описание	Пример
amount	Сумма	9.99
pan	Номер карты	4111111111111111

Необязательные параметры

Необязательные параметры те же, что и в [Авторизации](#).

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -X POST -H 'Authorization: Basic cHJvamVjdDpwYXNzd29yZA==' -i 'http://project:***@api.box:5001/orders/credit' --data '{
```

```
{
  "amount"      : "99.99",
  "pan"         : "4111111111111111"
}'
```

```
> HTTP/1.1 200 OK
> Server: nginx/1.10.3 (Ubuntu)
> Date: Tue,
> 18 May 2021 10: 37: 12 GMT
> Content-Type: application/json
> Content-Length: 786
> Connection: keep-alive
>
```

```
{
  "orders": [
    {
      "descriptor": " merchant123",
      "amount": "99.99",
      "status": "credited",
      "client": {},
      "amount_refunded": "0.00",
      "description": null,
      "operations": [
        {
          "auth_code": "301541",
          "cashflow": {
            "currency": "USD",
            "fee": "0.00",
            "incoming": "-99.99",
            "reserve": "3.00",
            "receivable": "-102.99",
            "amount": "-99.99"
          },
          "status": "success",
          "amount": "99.99",
          "iso_response_code": "00",
          "currency": "USD",
          "created": "2021-05-18 13:37:12",
          "type": "credit",
          "iso_message": "Approved"
        }
      ]
    }
  ]
}
```

```

    ],
    "pan": "411111****1111",
    "secure3d": {},
    "created": "2021-05-18 13:37:11",
    "custom_fields": {},
    "currency": "USD",
    "card": {
      "type": "mastercard",
      "subtype": "standard",
      "expiration": "0/0"
    },
    "id": "61277755379729234",
    "issuer": {
      "bin": "527644"
    },
    "amount_charged": "0.00",
    "location": {},
    "auth_code": "301541",
    "merchant_order_id": null,
    "updated": "2021-05-18 13:37:12"
  }
]
}

```

Завершение 3-D Secure аутентификации (POST /orders/:id/complete)

Обязательные параметры

Название	Описание	Пример
PaRes	Ответ системы аутентификации (ACS)	424EF64895CAE9797497F

Пример:

```

curl -X POST 'http://project:***@api.box:5001/orders/43809080318125930/complete' -H 'accept: application/json' -H 'authorization: Basic bWVyY2hhbnRfdGVzdDpQQVFpemtWeCtDUlJVbTEr' -H 'Content-Type: application/json' -d '{"PaRes": "YAED53A345D"}'

```

```

{
  "orders": [
    {
      "description": "Book sale 453",
      "auth_code": "AUTH12",
      "amount": "100.99",
      "merchant_order_id": null,
      "id": "43809080318125930",
      "updated": "2019-03-20 18:47:03",
      "pan": "411111****1111",
      "currency": "USD",
      "descriptor": "321/123",
      "amount_charged": "100.99",
      "created": "2019-03-20 18:46:15",
      "status": "charged",
      "amount_refunded": "0.00"
    }
  ]
}

```

Примечание

Эндпойнт передает ответ на запрос авторизации. При повторной отправке запроса на эндпойнт будет

получен ответ: "Order already completed".

Завершение 3-D Secure 2.0 аутентификации (POST /orders/:id/complete3d20)

Необязательные параметры

Название	Описание	Пример
<code>cres</code>	Base64 encoded строка, содержащая результаты 3-D Secure аутентификации	<code>ewogICJhY3NUcmFuc0...W9uI iA6ICiYlJlEuMCIKfQ==</code>

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```
curl -X POST -H 'Authorization: Basic cHJvamVjdDpwYXNzd29yZDEyMw==' -i  
'http://project:***@api.box:5001/orders/57937457169525773/complete3d20'  
  
{  
  "orders": [{  
    "merchant_order_id": null,  
    "client": {},  
    "secure3d": {  
      "reason": "force3d",  
      "xid": "635fb7bc-7acc-451e-a37d-606f1febae78",  
      "eci": "5",  
      "version": "2",  
      "authorization_status": "Y",  
      "dsrp": "0",  
      "cavv": "AAABBZEEBgAAAAAAAAQGAAAAAAAA=",  
      "scenario": "full"  
    },  
    "updated": "2020-12-18 13:13:43",  
    "amount": "123.00",  
    "currency": "USD",  
    "status": "authorized",  
    "pan": "510033****0101",  
    "amount_refunded": "0.00",  
    "amount_charged": "0.00",  
    "operations": [{  
      "amount": "123.00",  
      "auth_code": "123",  
      "cashflow": {  
        "currency": "USD",  
        "receivable": "-3.69",  
        "amount": "0.00",  
        "fee": "0.00",  
        "incoming": "0.00",  
        "reserve": "3.69"  
      },  
      "type": "authorize",  
      "iso_message": "Approved",  
      "created": "2020-12-18 13:13:43",  
      "iso_response_code": "00",  
      "status": "success",  
      "currency": "USD"  
    }],  
    "auth_code": "123",  
    "description": null,  
    "custom_fields": {}  
  }]
```

```

"location": {
  "ip": "6.6.6.6"
},
"card": {
  "type": "mastercard",
  "expiration": "10/22",
  "subtype": "standard"
},
"issuer": {
  "bin": "510033"
},
"id": "57937457169525773",
"created": "2020-12-18 13:09:52"
}]
}

```

Примечание

Эндпойнт завершает авторизацию, проведенную с аутентификацией 3-D Secure 2.0. Если в свойствах заказа нет данных идентификации из *RReq*, будет получена ошибка валидации и заказ останется в статусе *prepared*.

Продолжение 3DS Method URL аутентификации (POST /orders/:id/resume)

Параметры

Не принимаются.

Ответ

Поля ответа повторяют набор полей из раздела [Информация о заказе](#).

Пример:

```

curl -X POST -H 'Authorization: Basic cHJvamVjdDpwYXNzd29yZDEyMw==' -i
'http://project:***@api.box:5001/orders/60391976929806541/resume'

{
  "orders": [{
    "amount": "1.00",
    "custom_fields": {},
    "id": "60391976929806541",
    "form3d": {
      "threeDSSESSIONData": "NjAzOTE5NzY5Mjk4MDY1NDE=",
      "method": "POST",
      "action": "http://checkout.***.local:80/complete/secure3d20"
    },
    "issuer": {
      "bin": "220138"
    },
    "amount_refunded": "0.00",
    "amount_charged": "0.00",
    "created": "2021-04-08 12:29:19",
    "form3d_html": "<form id=\"form3d\" method=\"POST\"
action=\"http://checkout.***.local:80/complete/secure3d20\"><input type=\"hidden\"
name=\"threeDSSESSIONData\" value=\"NjAzOTE5NzY5Mjk4MDY1NDE=\"></form><script>window.onload =
function() {document.forms['form3d'].submit();}</script>",
    "updated": "2021-04-08 12:29:20",
    "status": "prepared",
    "description": null,
  }
]
}

```

```

    "location": {
      "ip": "6.6.6.6"
    },
    "currency": "EUR",
    "client": {},
    "pan": "220138****0013",
    "merchant_order_id": null,
    "operations": [],
    "card": {
      "holder": "John Smith",
      "type": "mir",
      "subtype": "unknown",
      "expiration": "12/21"
    },
    "secure3d": {
      "dsrp": "0",
      "scenario": "unknown",
      "reason": "force3d",
      "eci": "0",
      "version": "2"
    }
  }
}

```

Примечание

Эндпойнт продолжает авторизацию, описанную в разделе [Платежи с 3-D Secure 2.0](#). Максимальное время ожидания ответа на запрос - 10 секунд.

Дополнительные возможности

Динамический дескриптор

Эта опция доступна, если ее поддерживает банк-эквайер.

Динамический дескриптор позволяет передавать имя и местоположение продавца, которые попадут в банковскую выписку держателя карты и будут указаны в SMS, посылаемом банком эмитентом.

Название и местоположение продавца, установленные при регистрации торговой точки (MID) в банке, можно переопределять актуальным значением для каждого платежа. Динамический дескриптор поддерживается в операциях [Authorize](#), [Create](#) и [Rebill](#).

Данные динамического дескриптора должны быть указаны в блоке `extra_fields`:

Обязательные параметры

Название	Описание	Пример
<code>dynamic_descriptor</code>	Динамический дескриптор (*)	—
<code>*.merchant</code>	Имя продавца	BOOKSTORE LTD
<code>*.city</code>	Город продавца	Moscow
<code>*.country</code>	Страна продавца (ISO 3166-1 alpha-2)	RU

Согласно значениям, приведенным в примере, дескриптор, полученный владельцем карты, может выглядеть следующим образом:

BOOKSTORE LTD>Moscow

RU

Примечание

Суммарная длина значений полей *merchant* и *city* не должна превышать 37 символов, при этом длина значения *merchant* не должна превышать 22 символов. Значения обоих параметров могут содержать только цифро-буквенные символы и знаки препинания.

Поддержка длинной записи (Airline Addendum)

Для идентификации платежей, связанных с заказом авиабилетов, есть возможность передавать длинную запись в запросах [Create](#), [Authorize](#), [Rebill](#), [Charge](#).

Длинная запись передается эквайеру при списании средств. Если в запросе [Charge](#) указана длинная запись, отличающаяся от той, что была указана при авторизации или создании заказа, то она будет переопределена.

Длинная запись передается в блоке *extra_fields*:

Обязательные параметры

Название	Описание	Пример
<code>airline_addendum</code>	Длинная запись (*)	—
<code>*.raw_data</code>	Данные длинной записи	01QP1INU1237...3JOHNDOO

Уникальная конверсия платежей

Если при платеже передается уникальный идентификатор заказа (инвойса), все заказы, имеющие один и тот же идентификатор, при анализе конверсии на платежной странице могут считаться одной попыткой. Это позволяет более точно учитывать конверсию в тех случаях, когда покупатель делает несколько попыток платежей разными картами, чтобы оплатить один заказ.

Уникальный идентификатор заказа можно передавать в запросах [Create](#), [Authorize](#), [Rebill](#).

Идентификатор передается в блоке *extra_fields*:

Название	Описание	Пример
<code>attributes</code>	Параметры (*)	—
<code>*.invoice_id</code>	Уникальный идентификатор заказа	741852963

Платежи с 3-D Secure 2.0

Данные для проведения платежа с 3-D Secure 2.0 аутентификацией передаются в запросе [Authorize](#) в параметре *secure3d*.

Обязательные параметры

Название	Описание	Пример
<code>secure3d</code>	Данные аутентификации 3-D Secure (*)	—
<code>*.browser_details</code>	Параметры браузера	—
<code>*.*.browser_accept_header</code>	Типы принимаемого контента	text/html
<code>*.*.browser_color_depth</code>	Глубина цвета устройства	24
<code>*.*.browser_ip</code>	IP пользователя	109.86.86.86
<code>*.*.browser_language</code>	Язык браузера	en
<code>*.*.browser_screen_height</code>	Высота области браузера	1080
<code>*.*.browser_screen_width</code>	Ширина области браузера	1920
<code>*.*.browser_timezone</code>	Разница между таймзоной браузера и временем UTC в минутах	-180
<code>*.*.browser_user_agent</code>	Юзер агент браузера	Mozilla

<code>*.*browser_java_enabled</code>	Флаг включения в браузере поддержки Java	false
<code>*.*window_height</code>	Высота окна браузера	1080
<code>*.*window_width</code>	Ширина окна браузера	1920

Необязательные параметры

Название	Описание	Пример
<code>secure3d</code>	Данные аутентификации 3-D Secure (*)	—
<code>*.force_challenge</code>	Принудительное включение сценария challenge flow	1

Если ответ на авторизацию имеет такой вид:

```
{
  "method3d_params": {
    "method_url": "https://ds.testcert.testconnect.su/ma",
    "method_data":
"axJ0aHJ1ZURTfWV0aG9kTm90aWZpY2F0aW9uVWJMIjoiaHR0cHM6Ly8zZHN0ZXN0Lm1kbWJhbmsucnUvY2dpLWJpb
i9jZ21fbGlua193IiwidGhyZWVEU1NlcnZlc1RyYW5zSUQiOiJkNDI0ZDFjMS02OTUwLTRkNmMtYTMwNy0zMTCxZDgzYjJiN2IifQ"
  },
  "message": "Method3d required",
  "order_id": "60391976929806541"
}
```

то необходимо собрать в браузере пользователя форму, которая отправляет данные в невидимый *iframe*, используя для него значения параметров `method_url` и `method_data` из ответа на авторизацию.

После отправки браузером пользователя формы в невидимый *iframe*, для продолжения авторизации необходимо выполнить запрос, описанный в разделе [Продолжение 3DS Method URL аутентификации](#).

DSRP-платежи

Данные DSRP-платежа передаются в параметре `secure3d`

Обязательные параметры

Название	Описание	Пример
<code>secure3d</code>	Данные аутентификации 3-D Secure (*)	—
<code>*.dsrp</code>	Признак DSRP-платежа	1
<code>*.xid</code>	Идентификатор транзакции из PaRes (XID)	8Rg2xYbxLUGsfZ1koh1JxAQ FBM=
<code>*.cavv</code>	Значение CAVV из PaRes	AAABCBVRA2EYZXQBV1EDAAA AAAA=
<code>*.eci</code>	Признак ECI	05

Необязательные параметры

Название	Описание	Пример
<code>dsrp_type</code>	Тип платежной системы DSRP. Доступные типы: <code>masterpass</code> , <code>apple_pay</code> , <code>android_pay</code> , <code>samsung_pay</code>	masterpass
<code>dsrp_wallet_id</code>	ID цифрового кошелька	765

Параметр `dsrp_wallet_id` применяется только если `dsrp_type == masterpass`. Если тип платежной системы не указан, по умолчанию считается, что это ApplePay.

При работе с DSRP-платежами мерчант получает токен, в котором содержатся реквизиты для проведения платежа. Заполнение параметров на примере ApplePay:

Название в токене	Название в API
<code>applicationPrimaryAccountNumber</code>	<code>pan</code>
<code>applicationExpirationDate</code>	<code>expiration_month expiration_year</code>
<code>transactionAmount</code>	<code>amount</code>
<code>transactionId</code>	<code>xid</code>
<code>onlinePaymentCryptogram</code>	<code>cavv</code>
<code>eciIndicator</code>	<code>eci</code>

Поля `holder` и `cvv` не передаются, т.к. они не содержатся в токене, `dsrp` всегда имеет значение 1 (`true`), `eci` передается только, если содержится в токене.

Привязка платежной карты (Oneclick)

Привязка платежной карты на платежной странице дает возможность держателю сохранить данные своей карты при первом платеже, для того чтобы при последующих платежах он смог выбрать эту карту из списка своих привязанных карт и провести платеж с вводом только CVV-кода. Для того, чтобы карта привязалась, пользователь должен отметить соответствующий чекбокс на платежной странице.

Чтобы задействовать механизм привязки карты покупателя, в запросе создания ордера необходимо указать уникальный идентификатор пользователя в системе мерчанта. Если не передать этот идентификатор, пользователь увидит обычную платежную страницу без сохраненных карт и без опции сохранения введенной карты.

Идентификатор передается в блоке `extra_fields` запроса [Create](#), допускается передавать цифробуквенное значение длиной до 40 символов.

Название	Описание	Пример
<code>oneclick</code>	Привязка карты (*)	—
<code>*.customer_id</code>	Идентификатор покупателя	0123654789
<code>*.prechecked</code>	Флаг заполнения чекбокса сохранения карты	1

Постраничная выгрузка списков ордеров и операций

Для списка заказов или операций можно задать количество записей на странице и указать номер страницы для выгрузки.

Название	Описание	Пример
<code>page</code>	Номер выгружаемой страницы	2
<code>page_size</code>	Количество записей на странице	50

Если параметры `page` и `page_size` не указаны, то будет выгружена страница с номером 1, которая будет содержать количество записей по умолчанию.

Например, если `page = 2` и `page_size = 50`, то будет выгружено 50 записей, начиная с 51-ой.

HTTP-заголовки ответа содержат в заголовке `Pagination` ссылки на последующую и предыдущую страницы списка ордеров (операций). Если выгружаемая страница первая, то будет ссылка только на следующую страницу. Например, если выгружена вторая страница списка, то заголовок будет содержать ссылки такого вида:

```
'<http://mybox.localhost?foo=bar&page_size=2&page=1>; rel="prev",  
<http://mybox.localhost?foo=bar&page_size=2&page=3>; rel="next"'
```

Если будут установлены некорректные значения параметров `page_size` или `page`, то возникнет ошибка

валидации с таким сообщением:

```
'message' : 'Invalid page size' или 'message' => 'Invalid page'
```

Фискализация по 54-ФЗ

Общая информация

Система позволяет передать вместе с платежом данные для формирования электронного чека. Чек будет зарегистрирован с помощью одного из провайдеров фискализации и после оформления его реквизиты станут доступны для просмотра в бэкофисе мерчанта и в API.

Регистрация чека происходит только после успешной операции списания [Charge](#), оформление проводится асинхронно через некоторое время после регистрации.

В зависимости от используемой схемы интеграции, данные чека можно передать в запросах [Create](#), [Authorize](#), [Rebill](#), [Charge](#).

Если в интеграции не используется явное проведение списаний через API (одностадийная схема или автоклиринг), данные чека следует передавать в запросах на оплату (операции [Create](#), [Authorize](#), [Rebill](#)). Если используется явный вызов списания, данные чека можно передать непосредственно в операцию [Charge](#), при этом ранее переданные данные по этому платежу будут перезаписаны.

При оформлении возврата данные чека "Возврат прихода" передаются вместе с операцией [Refund](#) при проведении ее через API, независимо от схемы интеграции. Если провести операцию возврата через бэкофис, или не передать данные чека в API, возврат средств будет проведен без оформления электронного чека.

Регистрация чека

Данные для формирования чека следует передавать в блоке `extra_fields`:

Обязательные параметры

Name	Description	Example
<code>electronic_receipt</code>	Электронный чек (*)	—
<code>*.phone</code>	Телефон покупателя	9111231212
<code>*.email</code>	Е-mail покупателя	mail@server.example
<code>*.inn</code>	ИНН продавца	12345678
<code>*.payment_address</code>	Адрес места расчетов	shop.ru
<code>*.taxation_type</code>	Система налогообложения продавца	1
<code>*.items</code>	Список товарных позиций	—
<code>*.*.name</code>	Наименование	Книга
<code>*.*.price</code>	Цена	300.25
<code>*.*.quantity</code>	Количество	2
<code>*.*.amount</code>	Стоимость	600.50
<code>*.*.tax_type</code>	Ставка НДС	vat18

Оptionальные параметры товарной позиции `*.items`

Name	Description	Example
<code>*.*.unit</code>	Единица измерения товара	шт.
<code>*.*.payment_object</code>	Предмет расчета	goods
<code>*.*.payment_method</code>	Способ расчета	full_payment

<code>*.*.agent</code>	Атрибуты агента	—
<code>*.*.*.type</code>	Тип агента	attorney
<code>*.*.*.operation</code>	Операция платежного агента	Платеж
<code>*.*.*.phone_numbers</code>	Телефоны платежного агента (массив строк)	["79111231212", "74959999999"]

Опциональные параметры

Name	Description	Example
<code>customer</code>	Покупатель	Кох Василий
<code>customer_tax_number</code>	ИНН покупателя	897642804565

В параметре `*.items` передается массив структур с товарными позициями. В таблице перечислены поля для структуры с одной товарной позицией.

В параметре `*.taxation_type` указывается система налогообложения продавца в соответствии с таблицей ниже:

Значение	Система налогообложения
0	Общая
1	Упрощенная (доходы)
2	Упрощенная (доходы минус расходы)
3	Единый налог на вмененный доход
4	Единый сельскохозяйственный налог
5	Патентная

В параметре товарной позиции `*.*.tax_type` указывается ставка, по которой товар облагается НДС:

Значение	Описание
<code>none</code>	НДС не облагается
<code>vat0</code>	Ставка НДС 0%
<code>vat18</code>	Ставка НДС 18%
<code>vat10</code>	Ставка НДС 10%
<code>vat118</code>	Ставка НДС из расчета 18/118
<code>vat110</code>	Ставка НДС из расчета 10/110

В параметре товарной позиции `*.*.payment_object` указывается признак предмета расчета:

Значение	Описание
<code>goods</code>	Товары
<code>excisable_goods</code>	Подакцизные товары
<code>work</code>	Работа
<code>service</code>	Услуги
<code>gambling_bet</code>	Ставка азартной игры
<code>gambling_prize</code>	Выигрыш азартной игры
<code>lottery_ticket</code>	Лотерейный билет
<code>lottery_prize</code>	Выигрыш лотереи
<code>intellectual_property</code>	Предоставление результатов интеллектуальной деятельности
<code>payment</code>	Платеж
<code>agent_commission</code>	Агентское вознаграждение
<code>composite</code>	Составной предмет расчета
<code>other</code>	Другой предмет расчета

В параметре товарной позиции `*.*.payment_method` указывается признак способа расчета:

--	--

Значение	Описание
full_payment	Полная оплата
partial_payment	Частичная оплата
full_prepayment	Полная предоплата
partial_prepayment	Частичная предоплата
advance_payment	Аванс
credit	Передача в кредит
credit_payment	Оплата кредита

В параметре атрибутов агента для товарной позиции `*.*.agent.type` указывается тип платежного агента:

Значение	Описание
bank_paying_agent	Банковский платежный агент
bank_paying_subagent	Банковский платежный субагент
paying_agent	Платежный агент
paying_subagent	Платежный субагент
attorney	Поверенный
commission_agent	Комиссионер
other	Другой агент

Пример:

```
{
  ...
  "extra_fields":{
    "electronic_receipt":{
      "email":"test@mail.example",
      "items":[
        {
          "name":"Товар1",
          "amount":200,
          "price":200,
          "quantity":1,
          "tax_type":"vat18",
          "unit" : "шт.",
          "payment_object" : "goods",
          "payment_method" : "full_payment",
          "agent":{
            "type" : "attorney",
            "operation" : "Платеж",
            "phone_numbers" : ["79111231212","74959999999"]
          }
        },
        {
          "name":"Товар2",
          "amount":300,
          "price":300,
          "quantity":1,
          "tax_type":"vat18",
          "unit" : "час",
          "payment_object" : "service",
          "payment_method" : "full_prepayment"
        }
      ]
    },
    "payment_address":"testshop.example",
    "taxation_type":"1",
    "inn":"7736642031",
    "phone":"+79111231212"
  }
}
```

```
}  
}  
}
```

Запрос реквизитов чека

Все чеки, связанные с ордером, передаются в ответ на запрос [Информации о заказе](#) в списке *electronic_receipts*. Каждый элемент списка содержит следующие поля:

Имя	Описание	Пример
<code>status</code>	Состояние	success
<code>operation</code>	Операция	refund
<code>created</code>	Дата создания	2017-11-01 18:41:15
<code>attributes</code>	Атрибуты (*)	—
<code>*.shift_receipt_number</code>	Номер чека в смене	2779
<code>*.shift_number</code>	Номер смены.	7971
<code>*.registered</code>	Дата и время документа из ФН.	2017-11-01 18:45:00
<code>*.fiscal_storage_number</code>	Номер ФН.	9999078900001341
<code>*.device_registration_number</code>	Регистрационный номер ККТ.	0000000400054952
<code>*.receipt_number</code>	Фискальный номер документа.	1290
<code>*.receipt_attribute</code>	Фискальный признак документа.	819694071
<code>*.fns_website</code>	Адрес сайта ФНС.	www.nalog.ru

Тестирование

Общая информация

Тестовые заказы проводятся только через тестовый терминал. Доступ к тестовому терминалу можно получить через службу поддержки.

В тестовом режиме допускается использовать любые тестовые номера карт с корректными датой истечения, секретным кодом и именем держателя. Тестовый терминал успешно проводит попытки оплаты с любыми реквизитами, которые прошли базовую валидацию формата.

Для получения определенного результата платежа можно использовать специальные номера карт:

Номер карты	Описание
4111111111111111	Успешная авторизация
2222400060000007	Успешная авторизация
4276990011343663	Без 3-D Secure: отклоненная авторизация;
	3-D Secure: сценарий "Unable to verify enrollment".
5555555555555599	Внутренняя ошибка эквайера или шлюза
4276838748917319	Сценарий 3-D Secure "Not enrolled"
4000000000000002	Отклонено системой защиты от мошеннических операций