

# ОПИСАНИЕ ПРОДУКТА

## МУЛЬТИРАСЧЕТЫ

06.11.2025



## Оглавление

История изменений .....	3
Термины и сокращения .....	4
1. Описание продукта .....	5
2. Настройка подключения .....	5
3. Варианты подключения .....	6
4. Статусная модель Сделки .....	8
4.1. Валидация сделки на стороне банка .....	9
4.2. Валидация сделки на стороне Сайта .....	10
5. Алгоритм формирования подписи запроса на выплаты .....	11
5.1. С помощью параметра Token .....	11
5.2. С помощью RSA сертификата .....	11
5.3. С помощью сертификата КриптоПро .....	12
6. Функционал N списаний : N выплат .....	13
6.1. Для партнера, использующего форму оплаты Банка (не PCI DSS) .....	13
6.2. Для партнера, использующего собственную форму оплаты (PCI DSS) .....	14
7. Метод createSpDeal .....	24
8. Метод closeSpDeal .....	25

## История изменений

Версия	Описание	Дата
1.0	Документ создан	12.08.2024
1.1	Обновлен тип параметра PartnerId в таблице 6.2	4.09.2024
1.2	Обновлены примеры запросов	24.09.2024
1.3	Обновлено описание LevelOfConfidence	19.05.2025
1.4	Обновлена обязательность параметра PaymentRecipientId	6.11.2025

## Термины и сокращения

Термин	Определение
Продавец	Зарегистрированный пользователь Сайта (Приложения), размещающий там Объявления с предложением заключить Сделку в отношении Товара с использованием Сервиса
Покупатель	Зарегистрированный пользователь Сайта (Приложения), осуществляющий просмотр размещенного Продавцом Объявления, взаимодействие с Продавцом в отношении Товара, заключивший с Продавцом Сделку с использованием Сервиса.
Товар	Товар, услуга (работа), в отношении которого Продавец размещает на Сайте (Приложении) Объявление, являющийся предметом Сделки.
Объявление	Информационное сообщение с предложением о Товаре (включая контактную информацию, фотографии и любую сопутствующую информацию), размещаемое Продавцом на Сайте/Приложении, адресованное неопределенному кругу лиц.
Сайт (Приложение, Площадка)	Интернет-ресурс, предоставляющий возможность размещения Продавцами Объявлений о продаже Товаров, а также предоставляющий Покупателям возможность поиска, просмотра предложений Продавцов с целью последующего приобретения Товара с использованием Сервиса.
Заказ	Запрос на Покупку размещенного Товара на Сайте/Приложении.
Покупка	Авторизация и списание средств с банковской карты Покупателя за покупку Товара.
Выплата Продавцу	Пополнение банковской карты или банковского счета Продавца за Покупку Товара, необходимо указать идентификатор Покупки. Осуществляется после подтверждение списания средств с банковской карты Покупателя (Покупки).
Сделка	Соглашение, заключенное между Покупателем и Продавцом, предметом которого является купля-продажа Товара с использованием функционала Сайта, и устанавливающее порядок выполнения сторонами Сделки (Покупателем и Продавцом) ее условий.

## 1. Описание продукта

"Мультирасчеты" — это продукт, позволяющий организовать взаиморасчеты между Покупателем (физическими лицами) и Продавцом (физическими или юридическими лицами) при продаже товаров или услуг через различные интернет площадки.

При этом взаимодействие между Покупателем и Продавцом происходит не напрямую между собой, а при участии Площадки - юридического лица, которое организует размещение Товара на своей витрине, помогает Продавцу найти Покупателя, дает возможность воспользоваться сервисом банка для оплаты за товар и для получения денежных средств Продавцом после реализации товара, а также организует арбитраж, в случае такой необходимости.

Площадка контролирует ход выполнения сделки и дает команду Банку на списание, а после успешного выполнение Продавцом своих обязанностей (например, после исполнения оговоренных услуг или работ, либо после успешной доставки товара) инициирует выплату Продавцу.

Площадка берет свою комиссию за осуществление "Мультирасчетов".

Для осуществления выплат ЮЛ необходимо зарегистрировать это ЮЛ согласно [документации](#).

## 2. Настройка подключения

Для подключения продукта "Мультирасчеты" необходимо подать заявку своему персональному менеджеру или указать это в комментарии при подаче короткой заявки на [tbank.ru/kassa](http://tbank.ru/kassa). Также необходимо указать максимальный срок сделки (срок, после которого при отсутствии подтверждения успешного завершения сделки со стороны площадки денежные средства возвращаются на карту Покупателя, либо, при наличии в сделке успешных выплат, выплачиваются в пользу Площадки).

### 3. Варианты подключения

Ниже представлена матрица возможных реализаций подключений данного продукта на основании протоколов Банка, а также указание минимальных необходимых методов для реализации данного продукта.

Тип операции	Покупка		Выплата Продавцу	
Тип подключения	NOT PCI DSS	PCI DSS	NOT PCI DSS	PCI DSS
	Если используется стандартная платежная форма на стороне Т-Банка	Если используется платежная форма на стороне Площадки	Если используется стандартная платежная форма на стороне Т-Банка	Если используется платежная форма на стороне Площадки
		Полное описание протокола		Полное описание протокола
API	<p><b>Init</b> - инициализация Сделки на стороне банка.</p> <p><b>Confirm</b> - списание средств с банковской карты Покупателя.</p> <p><b>Cancel</b> - частичный или полный возврат денежных средств Покупателю. Если не было списания, то денежные средства расходятся. Отмена/частичная отмена Сделки.</p>	<p><b>Init</b> - создание Сделки на стороне банка.</p> <p><b>FinishAuthorize</b> - запрос на Покупку (авторизация средств на банковской карте Покупателя).</p> <p><b>Confirm</b> - списание средств с банковской карты Покупателя.</p> <p><b>Cancel</b> - частичный или полный возврат денежных средств Покупателю. Если не было списания, то денежные средства расходятся. Отмена/частичная отмена Сделки.</p>	<p><b>AddCustomer</b> – создание Продавца</p> <p><b>AddCard</b> - получение URL на привязку карты Продавца на стороне банка.</p> <p><b>Init</b> - проверка возможности выплаты на карту или счет Продавца. Передается DealId ранее созданной Сделки.</p> <p><b>Payment</b> - пополнение карты или счета Продавца.</p> <p><b>GetSbpMembers</b> - получение списка идентификаторов банков, участвующих в СБП</p>	<p><b>Init</b> - проверка возможности выплаты на карту или счет Продавца. Передается DealId ранее созданной Сделки.</p> <p><b>Payment</b> - пополнение карты или счета Продавца.</p> <p><b>GetSbpMembers</b> - получение списка идентификаторов банков, участвующих в СБП</p>

Тип операции	Покупка		Выплата Продавцу	
	Привязка карты на стороне банка, рекуррентные платежи	Привязка карты на стороне банка, рекуррентные платежи	Привязка карты на стороне банка. Подпись запроса с помощью Token *	Привязка карты на стороне банка. Подпись запроса с помощью Token *
<b>Дополнительные возможности API</b>				

\*Для выбора подписи запросов с помощью Token при подключении сообщите об этом своему персональному менеджеру или укажите это в комментарии при подаче короткой заявки на [tbank.ru/kassa](http://tbank.ru/kassa). Для переключения подписи с УНЭП на подпись запроса с помощью Token напишите на [acq\\_help@tbank.ru](mailto:acq_help@tbank.ru)

## 4. Статусная модель Сделки



Статус сделки	Какие выполнены операции	Описание этапа
Открыта	Есть успешные операции покупки (платежи в статусе CONFIRMED). Нет успешных операций выплаты (Payout) с переданным признаком финальной выплаты (FinalPayout = true, сделка не закрыта методом CloseSpDeal).	Денежные средства списаны с Покупателя/нескольких Покупателей.
Частично отменена	Есть хотя бы одна операция Cancel на часть суммы сделки.	Была хотя бы одна операция отмены\возврата на часть суммы сделки.
Закрыта	Есть успешная операция выплаты (Payout) с переданным признаком финальной выплаты (FinalPayout = true). или же сделка закрыта методом CloseSpDeal.	Денежные средства выплачены на карту Продавца.

#### 4.1. Валидация сделки на стороне банка

На стороне банка реализованы следующие проверки Сделки:

- запрос на проверку возможности выплаты на карту Продавца/запрос на завершение Сделки может быть только по ранее созданной Сделке (указан DealID ранее созданной Сделки);
- запрос на проверку возможности выплаты на карту Продавца может быть только если совершено списание денежных средств с банковской карты Покупателя, есть положительная сумма остатка Сделки;
- запрос на завершение Сделки может быть только по не закрытым сделкам;
- запрос на завершение Сделки может быть осуществлен только один раз в рамках одной Сделки;
- запрос на проверку возможности закрытия Сделки/запрос на завершение Сделки может быть только в пределах установленных лимитах на Сделку\*;
- сумма в запросе на проверку возможности выплаты на карту или счет Продавца должна быть меньше либо равна сумме остатка сделки;
- запрос на частичный или полный возврат денежных средств Покупателю не может быть осуществлен после закрытия Сделки.

Дополнительная логика, реализованная на стороне банка:

- Банк автоматически подтверждает списание денежных средств с карты Покупателя по истечению 4-х дней по картам ПС Visa Electron и 7-и дней по картам других ПС (MasterCard, МИР, Visa);
- Банк автоматически отменяет Заказ по истечению x-дней\*\*, если не поступило запроса на завершение Сделки.

## 4.2. Валидация сделки на стороне Сайта

При реализации продукта на своей стороне рекомендуется повторить логику валидации, осуществляющуюся на стороне банка во избежание низкой конверсии.

\* лимиты согласовываются с персональным менеджером

\*\* настройка на стороне Банка по требованиям Сайта. По умолчанию составляет 60 дней от даты авторизации денежных средств на карте Покупателя.

## 5. Алгоритм формирования подписи запроса на выплаты

### 5.1. С помощью параметра Token

Инструкция по формированию Token доступна [по ссылке](#)

В запросе на выплаты не участвуют параметры X509SerialNumber, DigestValue, SignatureValue.

Пример запроса в методе Init:

```
{
  "TerminalKey": "TinkoffBankTest",
  "Amount": "1751",
  "OrderId": "autoOrd1615285401068DELB",
  "CardId": "70000000857",
  "Token": "76956ae79e4b33e9833ab09c344c638e14d772d0e423aec4db65039622b2f1b6"
}
```

### 5.2. С помощью RSA сертификата

Для формирования подписи запроса необходимо:

1) Собрать массив всех передаваемых параметров в виде пар Ключ-Значение (кроме параметра DigestValue, SignatureValue, X509SerialNumber).

Например:

```
[{"TerminalKey", "TinkoffBankTest"}, {"PaymentId", "20150"}]
```

2) Сортировать по Ключам:

```
[{"PaymentId", "20150"}, {"TerminalKey", "TinkoffBankTest"}]
```

3) Конкатенировать значения:

20150 TinkoffBankTest

4) Вычислить хэш-сумму по алгоритму SHA256 и записать значение в DigestValue (должно получиться значение в Base64).

5) Декодировать DigestValue из Base64, подписать получившееся значение с помощью RSA ключа\* и записать в SignatureValue (должно получиться значение в Base64).

\* Инструкция по получению RSA ключа доступна [по ссылке](#).

Ниже представлены примеры реализации работы с библиотекой:

Язык программирования	Ссылка
Java	<a href="https://cdn.t-static.ru/static/documents/rsa-crypto-lib-java-mapi.zip">https://cdn.t-static.ru/static/documents/rsa-crypto-lib-java-mapi.zip</a>

### 5.3. С помощью сертификата КриптоПро

Для подписи запросов/ответов возможно пользоваться усиленной неквалифицированной электронной подписи (УНЭП) на алгоритмах ГОСТ и предоставить сертификат ключа проверки данной подписи в АО «Тинькофф Банк».

Для формирования подписи запроса необходимо:

- 1) Собрать массив всех передаваемых параметров в виде пар Ключ-Значение (кроме параметра DigestValue, SignatureValue, X509SerialNumber).

Например:

`[{"TerminalKey": "TinkoffBankTest"}, {"PaymentId": "20150"}]`

- 2) Сортировать по Ключам:

`[{"PaymentId": "20150"}, {"TerminalKey": "TinkoffBankTest"}]`

- 3) Конкатенировать значения:

`20150 TinkoffBankTest`

- 4) Вычислить хэш-сумму по ГОСТ Р 34.11-2012 256 и записать значение в DigestValue (должно получиться значение в Base64).

- 5) Декодировать DigestValue из Base64, подписать получившееся значение по ГОСТ Р 34.10-2012 256 и записать в SignatureValue. (должно получиться значение в Base64).

Для получения УНЭП можно обратиться к вашему менеджеру по взаимодействию, написав письмо с темой «Получение УНЭП\_Наименование организации».

В тексте письма указать:

Наименование системы: ЕАСQ (тест/прод).

Цель использования сертификата в системе: подпись методов протокола Интернет-Эквайринга.

Ниже представлены примеры реализации работы с библиотекой КриптоПро (CryptoPro):

Язык программирования	Ссылка	Версия КриптоПро
C#	<a href="#">Поддержка 2012 ГОСТ</a>	КриптоПРО CSP
Java	<a href="#">Поддержка 2012 ГОСТ</a>	КриптоПРО JCP
PHP	<a href="#">Поддержка 2012 ГОСТ</a>	КриптоПРО CSP

## 6. Функционал N списаний : N выплат

### 6.1. Для партнера, использующего форму оплаты Банка (не PCI DSS)

1. Площадка в первом платеже в рамках сделки в методе /v2/Init помимо параметров платежа передает параметр CreateDealWithTipe со значением 'NN' (вне блока DATA).
2. В ответ на запрос площадка получается ссылку на форму оплаты. Оплата проходит стандартно.
3. После успешного холдирования средств Покупателя Площадке отправляется http(s)нотификация о результатах выполнения данного действия. Нотификация будет содержать параметр идентификатора сделки – SpAccumulationId. Пример см. ниже.
4. После успешного оказания услуги Покупателю Площадка отправляет /v2/Confirm для списания средств с Покупателя. Запрос формируется стандартно без дополнительных параметров в контексте Сделки.
5. Для выплаты Продавцу площадка в метод /e2c/v2/Init передает необходимую сумму, которая не должна превышать сумму списаний по платежам, и параметр DealId. Если выплата является последней в сделке, в запросе должен быть также передан параметр FinalPayout со значением true (вне блока DATA).
6. В метод /e2c/v2/Payment передается PaymentId, полученный в ответ на вызов метода /e2c/v2/Init.
7. Выплата Продавцу производится на сумму, которая была отправлена в методе /e2c/v2/Init.
8. Для повторных выплат повторяются пп. 5-6. Повторные выплаты возможны в рамках остатка на балансе сделки. Повторные выплаты возможны, только если выполнены все условия одновременно:
  1. В предыдущих выплатах не был передан параметр FinalPayout со значением true,
  2. Сделка не закрыта методом closeSpDeal,
  3. На момент совершения выплаты баланс сделки должен превышать сумму выплат.
  4. Не истек срок жизни сделки.После передачи параметра FinalPayout со значением true, либо отправки метода closeSpDeal сделка закрывается.
9. Для открытия новой сделки Площадка в методе /v2/Init передает параметр CreateDealWithTipe='NN' и не передает параметр DealId. Одновременно может быть открыто любое количество сделок.

**Внимание!** Для сделок действует ограничение по времени жизни сделки, указанное в договоре п.1.7.11 или доп. соглашении. По истечении этого срока:

1. Все оплаты по сделке отменяются, если не было выплат.
2. Сделка закрывается с выплатой оставшейся суммы возмещения площадке, если по сделке была хотя бы одна выплата.

## 6.2. Для партнера, использующего собственную форму оплаты (PCI DSS)

1. Площадка в первом платеже в рамках сделки в методе /v2/Init помимо параметров платежа передает параметр CreateDealWithType со значением 'NN' (вне блока DATA).
2. После создания сделки в ответе будет возвращен идентификатор сделки DealId.
3. Если платеж проводится без проверки 3DS, то идентификатор сделки DealId возвращается в ответе метода /v2/FinishAuthorize.
4. Если платеж проводится с проверкой 3DS, то идентификатор сделки DealId возвращается в ответе метода v2/Submit3DSAuthorization для 3DS 1.0 или в ответе метода /v2/Submit3DSAuthorizationV2 для 3DS 2.0

Также идентификатор сделки возвращается как параметр SpAccumulationId в http(s)нотификациях соответствующих запросов.

5. После успешного оказания услуги Покупателю Площадка отправляет /v2/Confirm для списания средств с Покупателя. Запрос формируется стандартно без дополнительных параметров в контексте Сделки.
6. Для выплаты Продавцу площадка в метод /e2c/v2/Init передает необходимую сумму, которая не должна превышать сумму списаний по платежам, и параметр DealId. Если выплата является последней в сделке, в запросе должен быть также передан параметр FinalPayout со значением true (вне блока DATA).
7. В метод /e2c/Payment передается PaymentId, полученный в ответ на вызов метода /e2c/v2/Init. Выплата Продавцу производится на сумму, которая была отправлена в методе /e2c/v2/Init

8. Для повторных выплат повторяются пп. 6-7. Повторные выплаты возможны в рамках остатка на балансе сделки. Повторные выплаты возможны, только если выполнены все условия одновременно:
  1. В предыдущих выплатах не был передан параметр FinalPayout со значением true,
  2. Сделка не закрыта методом closeSpDeal,
  3. На момент совершения выплаты баланс сделки должен превышать сумму выплаты.
  4. Не истек срок жизни сделки.

После передачи параметра FinalPayout со значением true, либо отправки метода closeSpDeal сделка закрывается.

9. Для открытия новой сделки Площадка в методе /v2/Init передает в секции DATA параметр CreateDealWithType ='NN' и не передает параметр DealId. Одновременно может быть открыто любое количество сделок

**Внимание!** Для сделок действует ограничение по времени жизни сделки, указанное в договоре п.1.7.11 или доп. соглашении. По истечении этого срока:

1. Все оплаты по сделке отменяются, если не было выплат.
2. Сделка закрывается с выплатой оставшейся суммы возмещения площадке, если по сделке была хотя бы одна выплата.

## Пример передачи параметров на примерах запросов

/v2/Init метод POST Content-Type: json	e2c/v2/Init метод POST Content-Type: json
<pre>{   "TerminalKey": "TerminalKey",   "Amount": "15000",   "OrderId": "sp {\$timestamp}",   "Description": "Мультирасчеты",   "DATA": {     "Phone": "+71234567777",     "Email": "a@test.com"   },   "PaymentRecipientId": "asdasdad",   "DealId": "23123123",   "CreateDealWithType": "NN",   "LevelOfConfidence": "moderate",   "Token": "d0815e288f12131fb486cc5e9f91914a6d92c" }</pre>	<pre>{   "TerminalKey": "TerminalKeyE2C",   "OrderId": "autoOrd1615285401068DELb",   "CardId": "c0ad1dfc4e94ed4473",   "Amount": 1751,   "PaymentRecipientId": "asdasdad",   "DealId": "23123123",   "LevelOfConfidence": "moderate",   "Token": "d0815e288f12131fb486cc5e9f91914a6d92c",   "FinalPayout": true,   "senderAccountInfo": {     "addressInfo": {       "address": "Moscow 5 259",       "apartment": "259",       "building": "5",       "city": "Moscow",       "country": "643",       "street": " Moscow",       "postalCode": "215010"     },     "personInfo": {       "firstName": "Nikolay",       "lastName": "Petrov",       "middleName": "Test",       "dateOfBirth": "11.01.1990",       "citizenship": "RUS"     },     "passportInfo": {       "series": "00 11",       "number": "123456",       "issueDate": "11.01.2008"     },     "recipientAccountInfo": {       "personInfo": {         "firstName": "Petr",         "lastName": "Ivanov",         "middleName": " Test",         "dateOfBirth": "01.01.1988",         "citizenship": "BLR"       }     }   } }</pre>

e2c/v2/Init метод POST Content-Type: json Пример запроса на выплату по СБП	e2c/v2/Init метод POST Content-Type: json Пример запроса на выплату ЮЛ
<pre>{   "TerminalKey": "TerminalKeyE2C",   "OrderId": "testSBP 10",   "Phone": "79998887766",   "SbpMemberId": "100000000004",   "FinalPayOut": true,   "Amount": 100,   "DealId": "9043456",   "PaymentRecipientId": "79066589133",   "Token": "e24fd85c4c20e85d2eab5f65e8 b2066c83970b" }</pre>	<pre>{   "TerminalKey": "TerminalKeyE2C",   "OrderId": "TestOrder",   "Amount": 1751,   "PartnerId": "523456",   "DealId": "112233",   "FinalPayout": true,   "Token": "cb70d75fff815c433b17297593e66 eb33d0dd90ec5933b24272820cc564e7dca" }</pre>

Таблица 6.1 Описание параметров метода v2/Init

Наименование	Тип	Обязателен?	Описание
TerminalKey	String	Да	Идентификатор терминала. Выдается Мерчанту Т-Банк Бизнес при заведении терминала
Amount	Number	Да	Сумма в копейках. Например, сумма 3руб. 12коп. - это число 312 Минимальная сумма операции с помощью СБП составляет 10 руб.
OrderId	String	Да	Идентификатор заказа в системе Мерчанта
Token	String	Да	Подпись запроса
Description	String	Нет	Описание заказа. * Поле необходимо обязательно заполнять для осуществления привязки и одновременной оплаты по СБП. При оплате через СБП данная информация будет отображена в приложении мобильного банка клиента. Максимально допустимое количество знаков для передачи назначения платежа в СБП - 140 символов.

CustomerKey	String	Нет	Идентификатор клиента в системе Мерчанта. * Обязателен, если передан атрибут Recurrent. Если был передан в запросе, в нотификации будет указан CustomerKey и его CardId. Необходим для сохранения карт на платежной форме (платежи в один клик). Не является обязательным при рекуррентных платежах через СБП.
Recurrent	String	Нет	Для регистрации автоплатежа - обязательен. Если передается и установлен в Y, то регистрирует платеж как рекуррентный. В этом случае после оплаты в нотификации на AUTHORIZED будет передан параметр RebillId для использования в методе Charge.
PayType	String	Нет	Определяет тип проведения платежа – двухстадийная или одностадийная оплата. "О" - одностадийная оплата, "Т" - двухстадийная оплата  Если параметр передан - используется его значение. Если нет - значение в настройках терминала.
Language	String	Нет	Язык платежной формы. <ul style="list-style-type: none"> <li>• ru — русский</li> <li>• en — английский.</li> </ul> Если не передан, форма откроется на русском языке
NotificationURL	String	Нет	URL на веб-сайте Мерчанта, куда будет отправлен POST запрос о статусе выполнения вызываемых методов (настраивается в Личном кабинете): Если параметр передан – используется его значение. Если нет – значение в настройках терминала.
SuccessURL	String	Нет	URL на веб-сайте Мерчанта, куда будет переведен клиент в случае успешной оплаты: Если параметр передан – используется его значение. Если нет – значение в настройках терминала.
FailURL	String	Нет	URL на веб-сайте Мерчанта, куда будет переведен клиент в случае неуспешной оплаты: Если параметр передан – используется его значение. Если нет – значение в настройках терминала.

RedirectDueDate	String	Нет	<p>Срок жизни ссылки или динамического QR-кода СБП (если выбран данный способ оплаты). Если текущая дата превышает дату, переданную в данном параметре, ссылка для оплаты или возможность платежа по QR-коду становятся недоступными и платёж выполнить нельзя.</p> <p>* Максимальное значение: 90 дней от текущей даты.</p> <p>В случае, если параметр RedirectDueDate не был передан, проверяется настроенный параметр платежного терминала REDIRECT_TIMEOUT, который может содержать значение срока жизни ссылки в часах.</p> <p>Если его значение больше нуля, то оно будет установлено в качестве срока жизни ссылки или динамического QR-кода. Иначе устанавливается значение «по умолчанию» - 1440 мин. (1 сутки)</p> <p>Минимальное значение: 1 минута от текущей даты.</p> <p>Формат даты: YYYY-MM-DDTHH24:MI:SS+GMT</p> <p>Пример даты: 2016-08-31T12:28:00+03:00</p> <p>Если не передан, принимает значение 24 часа для платежа и 30 дней для счета</p>
PaymentRecipientId	String	Да	Идентификатор будущего получателя выплаты (номер телефона в формате "+79606747611").
DealId	String	Нет	Идентификатор сделки.
CreateDealWithType	String	Нет	Флаг о необходимости создания сделки при выполнении запроса. Необходимо заполнить значением «NN».
LevelOfConfidence	String	Нет	Уровень проверки получателя выплаты. Параметр необходим только в том случае, если его передачу потребовал отдел рисков на этапе подключения. Возможные значения: "low", "moderate", "high"

Таблица 6.2. Описание параметров метода e2c/v2/Init

Наименование	Тип	Обязателен?	Описание
TerminalKey	String	Да	Идентификатор терминала. Выдается Мерчанту Т-Банк Бизнес при заведении терминала
Amount	Number	Да	Сумма в копейках. Например, сумма 3руб. 12коп. — это число 312 Минимальная сумма операции с помощью СБП составляет 10 руб.
OrderId	String	Да	Идентификатор заказа в системе Мерчанта
Token	String	Да	Подпись запроса
CardId	String	Нет	Идентификатор карты пополнения, привязанной с помощью метода AddCard.
CardData	String	Нет	<p>Зашифрованные данные карты (используется только для Мерчантов с PCI DSS) Обязательный параметр для выплат на карту, если не был передан CardId. Объект CardData собирается в виде списка «ключ=значение» (разделитель «;»), зашифровывается открытым ключом (X509 RSA 2048), получившееся бинарное значение кодируется в Base64. Список значений:</p> <ol style="list-style-type: none"> <li>1. PAN - Номер карты. Число (обязательно)</li> <li>2. ExpDate - Месяц и год срока действия карты в формате MMYY. Число (обязательно)</li> <li>3. CardHolder - Имя и фамилия держателя карты (как на карте). Стока (опционально)</li> <li>4. CVV - Код защиты (с обратной стороны карты). Стока (опционально)</li> </ol> <p>Пример: PAN=43000000000000777;ExpDate=0523;CardHolder=IVAN PETROV;CVV=111</p> <p>Открытый ключ генерируется Т-Банком. Для получения открытого ключа для шифрования CardData можете написать на <a href="mailto:acq_help@tbank.ru">acq_help@tbank.ru</a> или обратиться к сотруднику, который вам помогал с</p>

			процессом выпуска терминала
PartnerId	String	Да*	ID партнёра магазина Мультирасчётов, которому предназначается выплата.*
Phone	String	Да*	Номер телефона получателя Формат: 11 цифр Пример: 70123456789
SbpMemberId	Number	Да*	Идентификатор банка-получателя в СБП Получить список идентификаторов банка можно через Метод GetSbpMembers
PaymentRecipientId	String	Да	Идентификатор получателя выплаты (номер телефона в формате "+71234567777")
DealId	String	Да	Идентификатор сделки.
FinalPayout	Boolean	Нет	Признак финальной выдачи. Если передан в значении true - сделка автоматически закроется после выплаты
LevelOfConfidence	String	Нет	Уровень проверки получателя выплаты. Параметр необходим только в том случае, если его передачу потребовал отдел рисков на этапе подключения. Возможные значения: "low", "moderate", "high"
senderAccountInfo	Object	Нет	Объект для передачи данных отправителя (необходим для выплат на иностранные карты)
recipientAccountInfo	Object	Нет	Объект для передачи данных получателя (необходим для выплат на иностранные карты)

Таблица 6.3. Структура объекта senderAccountInfo

Наименование	Тип	Обязателен?	Описание
AddressInfo	Object	Нет	Блок адресных данных отправителя
PersonInfo	Object	Нет	Блок персональных данных отправителя
PassportInfo	Object	Нет	Блок паспортных данных отправителя

Таблица 6.4. Структура объекта recipientAccountInfo

Наименование	Тип	Обязателен?	Описание
PersonInfo	Object	Нет	Блок персональных данных получателя

Таблица 6.5. Структура объекта AddressInfo

Наименование	Тип	Обязателен?	Описание
address	String	Нет	Адрес отправителя
apartment	String	Нет	Номер квартиры отправителя
building	String	Нет	Номер дома отправителя
city	String	Нет	Город отправителя
country	String	Нет	Код страны по ISO (из букв или цифр)
street	String	Нет	Улица отправителя
postalCode	String	Нет	Почтовый индекс отправителя

Таблица 6.6. Структура объекта PersonInfo

Наименование	Тип	Обязателен?	Описание
firstName	String	Нет	Имя
lastName	String	Нет	Фамилия
middleName	String	Нет	Отчество
dateOfBirth	String	Нет	Дата рождения
citizenship	String	Нет	Гражданство

Таблица 6.7. Структура объекта PassportInfo

Наименование	Тип	Обязателен?	Описание
series	String	Нет	Серия паспорта
number	String	Нет	Номер паспорта
issueDate	String	Нет	Дата выдачи паспорта

**Пример http(s)-нотификации с параметром SpAccumulationId:**

```
{  
  "TerminalKey": "TinkoffBankTest",  
  "OrderId": "test2",  
  "Success": true,  
  "Status": "CONFIRMED",  
  "PaymentId": "2006896",  
  "ErrorCode": "0",  
  "Amount": 102120,  
  "CardId": 867911,  
  "Pan": "430000**0777",  
  "ExpDate": "1122",  
  "SpAccumulationId": 756439112,  
  "Token": "d0815e288f121255d5d6b77831fb486cc5e9f91914a3f58a99b6118b54676d84"  
}
```

Параметр SpAccumulationId учитывается при формировании Token в нотификации.

## 7. Метод createSpDeal

**Описание:** Метод создает новую сделку.

### Запрос

Тестовый URL\*: <https://rest-api-test.tinkoff.ru/v2/createSpDeal>

Боевой URL: <https://securepay.tinkoff.ru/v2/createSpDeal>

\*Для возможности отправки запросов с IP-адресов, которые находятся вне РФ, напишите на почту [acq\\_help@tbank.ru](mailto:acq_help@tbank.ru) с просьбой добавить ваши IP в WL

Метод: POST

Таблица 7.1 Параметры запроса

Наименование	Тип	Обязателен?	Описание
TerminalKey	String	Да	Идентификатор оплатного терминала, выдается Площадке Банком
SpDealType	String	Да	Тип сделки (NN)
Token	String	Да	Подпись запроса

### Пример запроса:

```
{
  "TerminalKey": "TinkoffBankTest",
  "SpDealType": "NN",
  "Token": "30797e66108934dfa3d841b856fdad227c6b9c46d6a39296e02dc800d86d181e"
}
```

### Ответ

Таблица 7.2 Параметры ответа

Наименование	Тип	Обязателен?	Описание
SpAccumulationId	String	Да	Идентификатор сделки
Success	bool	Да	Успешность операции
ErrorCode	String	Да	Код ошибки, «0» - если успешно

### Пример ответа:

```
{
  "SpAccumulationId": "123",
  "Success": true,
  "ErrorCode": "0"
}
```

## 8. Метод closeSpDeal

**Описание:** Метод закрывает сделку при наличии операций по этой сделке. Все средства на балансе сделки на момент закрытия добавляются к сумме вознаграждения площадки. Выполняется на терминале оплаты, на котором была открыта сделка.

### Запрос

Тестовый URL\*: <https://rest-api-test.tinkoff.ru/v2/closeSpDeal>

Боевой URL: <https://securepay.tinkoff.ru/v2/closeSpDeal>

\*Для возможности отправки запросов с IP-адресов, которые находятся вне РФ, напишите на почту [acq\\_help@tbank.ru](mailto:acq_help@tbank.ru) с просьбой добавить ваши IP в WL

Метод: POST

Таблица 8.1. Параметры запроса

Наименование	Тип	Обязателен?	Описание
TerminalKey	String	Да	Идентификатор оплатного терминала, выдается Площадке Банком
SpAccumulationId	String	Да	Идентификатор сделки
Token	String	Да	Подпись запроса

### Пример запроса:

```
{
  "TerminalKey": "TinkoffBankTest",
  "SpAccumulationId": "111111",
  "Token": "30797e66108934dfa3d841b856fdad227c6b9c46d6a39296e02dc800d86d181e"
}
```

### Ответ

Таблица 8.2. Параметры ответа

Наименование	Тип	Обязателен?	Описание
Message	String	Нет	Краткое описание ошибки
Success	bool	Да	Успешность операции (true/false)
ErrorCode	String	Да	Код ошибки, «0» - если успешно

### Пример ответа:

```
{
  "Success": true,
  "ErrorCode": "0",
  "Message": "OK"
}
```