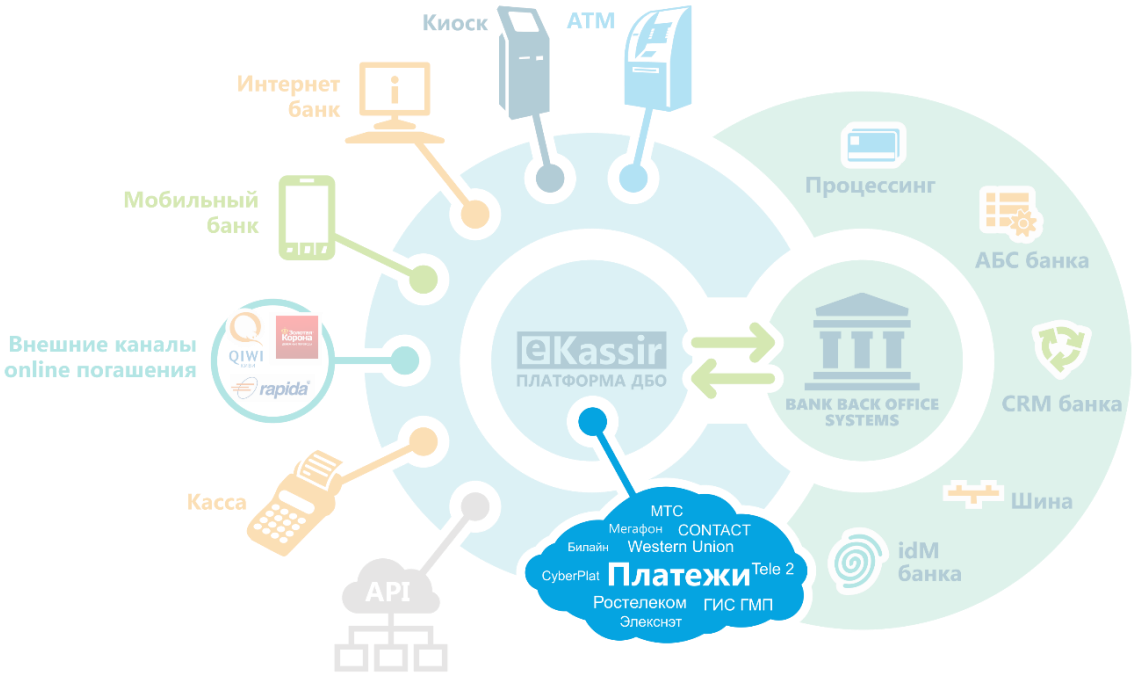


Merchant API. Методы для юрлиц.

Регистрация и получение QR-кодов.

Работа с реестрами платежей.

Инициирование возврата платежей.

Получение справочника участников

Методы для работы с платежными ссылками

Релиз 3.4

Санкт-Петербург 2025

Оглавление

1 История изменений документа 4

2 Введение 5

3 Авторизация API 5

3.1 Подключение кассового ПО с использованием логина и пароля 5

3.2 Подключение кассового ПО с использованием сертификата 6

3.3 Подключение POS-терминалов. Формирование Security Token с использованием приватного ключа, выданного УЦ банка 10

3.4 Подключение POS-терминалов. Формирование Security Token с использованием приватного ключа, выданного УЦ производителя терминала 12

3.5 Устаревший вариант авторизации 16

4 Точка входа 18

5 Регистрация и получение QR-кодов 18

5.1 Схема работы с динамическими QR-кодами 19

5.2 Формирование сообщения с содержимым application/jose 20

5.3 Получение QR-кода 21

5.3.1 Callbacks запроса 23

5.3.1.1 Получение платежа 23

5.3.1.2 Уведомление о привязке счета 24

5.3.1.3 Уведомление о финальном статусе 25

5.4 Запрос payload QR кода по идентификатору 26

5.5 Запрос статуса операций по Dynamic QR кодам 26

5.6 Запрос на выполнение платежа по привязанному счету 28

5.6.1 Callback запроса 29

5.6.1.1 Уведомление о разрешении на совершение платежа 29

6 Работа с реестрами платежей 29

6.1 Создание заявки для генерации реестра платежей 30

6.2 Запрос статуса готовности отчета по идентификатору заявки 31

7 Инициирование возврата платежей 32

7.1 Проверка. POST-запрос 32

7.2 Выполнение. PUT-запрос 34

7.3 Проверка. GET-запрос 36

8 Методы для получения справочника участников СБП 37

8.1 Получение списка участников СБП. GET- запрос 37

8.2 Получение данных участника СБП по идентификатору. GET-запрос 43

8.3 Получение логотипа участника СБП в виде файла. GET-запрос 44
9 Методы для работы с платежными ссылками 45
9.1 Регистрация платежной ссылки. POST - запрос 47
9.2 Изменение данных платежной ссылки. PUT - запрос 48
9.3 Активация кассовой ссылки. POST - запрос 50
9.4 Деактивация кассовой ссылки. DELETE- запрос 51

1 История изменений документа

Дата	Раздел	Изменение
19.09.2022	Регистрация кассовой платежной ссылки	В обработку запроса "Регистрация кассовой платежной ссылки" внесены изменения в проверки ТСП и Счета
20.10.2022	Получение QR-кода/Callback запроса	Добавлено описание параметра paramsId
05.12.2022	Методы для получения справочника участников СБП	Добавлена возможность фильтрации по ролям в рамках сценария При фильтре по сценарию появилась возможность использовать логические операторы И и ИЛИ
19.12.2024	Методы для работы с платежными ссылками	Обновлена диаграмма взаимодействия. Добавлены методы Запрос статуса Кассовой ссылки, Изменение данных для зарегистрированной Кассовой ссылки, Активация Кассовой ссылки V2, Статус Операции по Кассовой ссылке,

2 Введение

В этом документе приводятся примеры использования методов Merchant API для работы с QR-кодами, реестрами платежей и инициированием возвратов платежей.

Полное описание API находится в файлах *Merchant-API.yaml, B2C-Operation-Service.yaml* и *Members-API.yaml* в формате OpenAPI 3.0.

3 Авторизация API

3.1 Подключение кассового ПО с использованием логина и пароля

Процесс аутентификации

Инициатор	Получатель	Протокол	Описание	Передаваемые данные
Кассовое ПО	Адаптер СБП	HTTP 1.1 / OAuth2	Кассовое ПО выполняет запрос Access Token, используя username и password.	username password client_id
Адаптер СБП	Терминал	HTTP 1.1 / OAuth2	Access Manager проверяет находит пользователя по username, проверяет пароль, формирует Access Token и возвращает его в ответе	Access Token
Кассовое ПО	Адаптер СБП	HTTP 1.1 / OAuth2	Используя полученный Access Token терминал осуществляет запросы к Merchant API. Терминал контролирует время жизни Access Token, и при истечении запрашивает новый.	Access Token данные для запроса в API

Поля запроса Access Token заполняются следующим образом:

Поле	Обязательное	Описание	Источник данных
grant_type	Да	password	Константа
username	Да	Идентификатор системной учетной записи из Портала Агента	Настройка
password	Да	Пароль системной учетной записи из Портала Агента.	Настройка
scope	Да	Значение: *read*	Константа
client_id	Да	Идентификатор приложения, для которого должен быть выдан доступ Это - значение поля Application ID для Open ID Client, созданного для подключения кассового ПО. Open ID	Настройка

		Client создается в realm системных пользователей на сервере AM (раздел Applications)-Подробно о настройке realm - в документе «Портал Агента ТСП. Администрирование» (раздел «Создание и настройка области безопасности для системных пользователей»).	

Запрос:

```
POST https://ips-merchant-dev.ekassir.com/am/ipslegals/connect/token HTTP/1.1
Host: ips-merchant-dev.ekassir.com
Accept: application/json
X-Requested-With: XMLHttpRequest
X-Correlation-Id: CheckAuth
Request-Timeout: 30
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 89

grant_type=password&username=cashier&password=123456&scope=read&client_id= pos-terminal
```

Ответ:

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 15 Oct 2020 08:41:02 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Cache-Control: no-store, no-cache
Pragma: no-cache
Strict-Transport-Security: max-age=2592000
X-Correlation-Id: CheckAuth
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload


{"access_token":"eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1Q
zMxMzEiLCJ4NXQiOiJHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJuYmYiOjE2MDI3NTEy
NjIsImV4cCI6MTYwMjc1NDg2MiwiaXNzIjoiaHR0cHM6Ly9pcHMtbWVyY2hhbnQtZGV2LmVrYXNzaXIuY29tL2FtL2lwc2
xlZ2FscyIsImF1ZCI6Imh0dHBzOi8vaXBzLW1lcmNoYW50LWRldi5la2Fzc2lyLmNvbS9hbS9pcHNsZWdhbHMvcmVzb3Vy
Y2VzIiwiY2xpZW50X2lkIjoic3lzdXNlcnNsb2dpbiIsInNjb3BlIjoicmVhZCIsInN1YiI6IjViODE1MTIxLWYzZmQtNG
U0Yi1hNDVjLWFjNTQwMGFhOWU4NCIsImF1dGhfdGltZSI6MTYwMjc1MTI2MiwiaWRwIjoiZWthc3Npci5hbSIsInJvbGUi
OiJsZWdhbF9hZG1pbiIsImFtciI6WyJwYXNzd29yZCJdfQ.md-u1eY6Hzys8hkDOaIjFMpXrhWTb_-860DfRKzsBi-
9s50dzzD7A_IU_ctM9-3r4tRyN2SpwZynaJ0H5S3G3fxuD17Yy5x7PUJ9Jxk9mBVpNkt22L5y-
dbQdipFhguGHq4dsVukRl1avftxOi5u8eKWBILAtYfwGNk3ySubqtlEzwnPaTnbM-
Sog4peUJ9M8pojApWHUuw2JoXrQv4mQpaSiCE6myVj-97Tccrqdj2XGXsoPIw5boiqY0tRSyqOv7BYMjC-DIUu-
1iEtDVMfoFOzE2YDcxl4zAQiT_b6EfsO8_HU_W265jVTUvb2fHt-4G7NuQtmdz-
Z571Z78OiS1wLUpATqQR4LlKp9272sIImiKxvyQUPRCCMmQcKpzZaVTsw2rPrKyi4upG0GPL4vGZ4IK0j_CBqyZbu_EI_-
3llFK7QqzqcjCqskhhfgsPIBC8fCWB9oM58hMHsVYI8p8OVtxzqoUbblDmL7g4oYZmxcJLps2ZKn-
xo35jaPKF4gqGUHtE3JVCJHGtmIDVOItPrtGj_Qo0tgoPuro53jrpQhGB9T5CBddTuqADm4LlGi0V2XDMt_xeyxUfq0Q6O
F2B4wGxti7mUXlfAg07wLl3MkUPCazgAvRV5Z0yJZeS3DpANejcP5jtrOGt7Zuykje-
8FbambeFuDwm4EFXuYA","expires_in":3600,"token_type":"Bearer"}
```

3.2 Подключение кассового ПО с использованием сертификата

Процесс аутентификации

Инициатор	Получатель	Протокол	Описание	Передаваемые данные
Кассовое ПО	Адаптер СБП	HTTP 1.1 / OAuth2	• Кассовое ПО формирует Security Token и подписывает его приватным RSA ключом, выданным УЦ банка. • Кассовое ПО выполняет запрос Access Token, используя Security Token.	Security Token client_id
Адаптер СБП	Терминал	HTTP 1.1 / OAuth2	Access Manager проверяет подпись Security Token, проверяет наличие сервисной учетной записи, соответствующей используемому сертификату, формирует Access Token и возвращает его в ответе	Access Token
Кассовое ПО	Адаптер СБП	HTTP 1.1 / OAuth2	Используя полученный Access Token Кассовое ПО осуществляет запросы к Merchant API. ПО должно контролировать время жизни Access Token, и при истечении запрашивать новый.	Access Token данные для запроса в API

Формирование Security Token и Access Token

Терминал формирует токен безопасности (security token) и выполняет запрос токена доступа согласно разделу 6.3 стандарта RFC 7521 Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants. Для запроса используется токен безопасности в формате JSON Web Token (JWT) согласно разделу 2.1 стандарта JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants, утверждения которого заполняются следующим образом:

Утверждение (claim)	Обязательность Тип	Описание	Источник данных
Payload			
alg	да	Алгоритм шифрования. Поддерживаются RS256, RS384, RS512 и эллиптические кривые на NIST P-256, NIST P-384, NIST P-512.	
iss	да string	Issuer Для Security Token это официальный URL банка, либо сервиса СБП в банке Например: https://www.bank.ru/	Настройка
sub	да	Subject	Настройка

	string	Идентификатор системной учетной записи. В примере ниже это: "sub": "SystUser5082".	
aud	да string	Audience Содержит адрес token endpoint Access Manager. Access Manager отклоняет запросы с HTTP-статус кодом 400, в случае если утверждение заполнено неправильно. Например: https://ips-merchant-dev.ekassir.com/am/ipslegals/connect/token	Настройка
iat	нет integer	Issued At Дата и время выпуска токена в UTC. Access Manager отклоняет запрос, если Issued At в будущем. Например, 1516239022.	Вычисляется
exp	да integer	Expires At Дата и время срока действия токена в UTC. Access Manager отклоняет запрос, если Expires At в прошлом. Например, 1516240022.	Вычисляется, срок жизни токена задается в настройках
scope	Да string	Значение: *read* Не используется	
jti	нет string	JWT ID Уникальный идентификатор токена. При наличии данного утверждения Access Manager выполнит проверку уникальности идентификатора и отклонит запрос, если проверка неуспешна.	GUID, генерится при каждом запросе токена
Signature			
Подпись	да	Подпись JWT токена с использованием ключа длиной RSA 2048 байт и хэш-алгоритма SHA-256. Для подписи токена используется приватный, ключ выданный УЦ банка. Access Manager выполнит проверку подписи и отклонит запрос, если проверка неуспешна, с кодом ответа 400 и указанием в поле "error" кода ошибки "Token validation fault".	Вычисляется

Поля запроса access token заполняются следующим образом:

Поле	Обязательность	Описание	Источник данных
grant_type	Да	Всегда urn:ietf:params:oauth:grant-type:jwt-bearer	Константа
assertion	Да	Security Token в формате JWT	
scope	Да	Значение: *read*	
client_id	Да	Идентификатор приложения, для которого должен быть выдан доступ. Это - значение поля Application ID для Open ID Client, созданного для подключения кассового ПО. Open ID Client создается в realm системных пользователей на сервере АМ (раздел Applications)-Подробно о настройке realm - в документе «Портал Агента ТСП. Администрирование» (раздел «Создание и настройка области безопасности для системных пользователей»).	Настройка терминала

Пример claims:

Payload:

```
{
  "aud": "https://machinename.com/am/ipslegals/connect/token",
  "iss": "https://www.bank.ru/",
  "sub": "SystUser5082",
  "scope": read,
  "exp": 1602768994,
  "iat": 1602758194,
  "jti": "e50d9d3b33974ce4baa37b357ea23a53"
}
```

Запрос:

```
POST https://ips-merchant-dev.ekassir.com/am/ipslegals/connect/token HTTP/1.1
Host: ips-merchant-dev.ekassir.com
Accept: application/json
X-Requested-With: XMLHttpRequest
X-Correlation-Id: CheckAuth
Request-Timeout: 30
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 1251

grant_type=urn%3Aietf%3Aparams%3Aoauth%3Agrant-type%3Ajwt-
bearer&assertion=eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1Q
zMxMzEiLCJ4NXQiOiJHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJpc3MiOiJodHRwczov
L3d3dy5iYW5rLnJ1LyIsInN1YiI6IlN5c3RVc2VyNTA4MiIsImF1ZCI6Imh0dHBzOi8vaXBzLW1lcmNoYW50LWRldi5la2
Fzc2lyLmNvbS9hbS9pcHNsZWdhbHMvY29ubmVjdC90b2tlbiIsImlhdCI6MTYwMjc1ODIzNywiZXhwIjoxNjAyNzYxODM3
LCJzY29wZSI6InJlYWQiLCJqdGkiOiIxZTU3YWRjZWUwNzQ0YTNhODFiNzk4OTEyY2RmODVlMyIsIm5iZiI6MTYwMjc1OD
IzN30.W74Fydm3wlo_6w9-oocbt-xskgXTEliOb9EL7IdHXPZCBztfZSVYbW6SEQT4Z4LMt9zntW3jaToFPjb2A1_D-
Ze55iy_WwtxJWQzgObgyhgUroObXP5eHM5U-E5T_TY-
wAEXuJdKPrZA43WBE_8OLGeNCk05fXZoknG6svBcpco8FSFs0SUBv5GQ0VJFFduRfRZuFTWoa2f8D-vCKtJ4zvx5Dnyv-
rifgPmiJSyIJp967NMVhxUpL-
iB1rksNnzIikjq16KSCjuoUSfRwWfDFPs70IHV9Uci4iSo1KsEp_IxV0WPA9H9uqRbcrLqAL7jiCqNNC5XaoyGLQRZLvX9
wtex2mI6vp245XXnGoUaeYTHB9SrcDMoYGej5bUCaItUM9kaa-
```

```
4HXSSA1QIBSsJBp8suibpCkwUlW7NZql2hNaExgSPqosw0YYxya7Ou-
3p0TVvFTZG71HZfP24SZHJ4BtSxdI5isgPaMtMrovak9SDc82MPfdHf_s4_wVL29xe-
gapCsJQur7_oFJEGASMuj7bRmc1npIL5UJHeCHG0enDvuDGwFo_qgAEfP8nFc4K9n3QehUClQHoYZ2d-fsYjj9Q-
2683QfnUiFABgGTza30o4zaBH2GKFE7_FC9eW2XhmMuIsqJJckhGTTrs3Wvna0JGlEP-
D18gtuzlAkZSrN4&client_id=pos-terminal&scope=read
```

Ответ:

```
HTTP/1.1 200 OK
Server: nginx
Date: Wed, 14 Oct 2020 17:09:49 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Cache-Control: no-store, no-cache
Pragma: no-cache
Strict-Transport-Security: max-age=2592000
X-Correlation-Id: CheckAuth
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload
```


```
{"access_token":"eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1Q
zMxMzEiLCJ4NXQiOiJHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJuYmYiOjE2MDI3NTgy
MzksImV4cCI6MTYwMjc2MTgzOSwiaXNzIjoiaHR0cHM6Ly9pcHMtbWVyY2hhbnQtZGV2LmVrYXNzaXIuY29tL2FtL2lwc2
xlZ2FscyIsImF1ZCI6Imh0dHBzOi8vaXBzLW1lcmNoYW50LWRldi1la2Fzc2lyLmNvbS9hbS9pcHNsZWdhbHMvcmVzb3Vy
Y2VzIiwiY2xpZW50X2lkIjoicG9zLXRlcm1pbmFsIiwic2NvcGUiOiJyZWFkIiwic3ViIjoiU3lzdFVzZXI1MDgyIiwiYX
V0aF90aW1lIjoxNjAyNzU4MjM5LCJpZHAiOiJla2Fzc2lyLmFtIiwicm9sZSI6ImxlZ2FsX2FkbWluIiwibGVnYWxJZCI6
IkxBNDM4Njk5MjYzNyIsIm1lcmNoYW50SWQiOiJNQTcyNDg1Mjk1NDAiLCJhbXIiOlsidXJuOmlldGY6cGFyYW1zOm9hdX
RoOmdyYW50LXR5cGU6and0LWJlYXJlciJdfQ.uUyZBNpT2VNXothG9y77qq0SrkxsWmIzEfsgVt6RwwrNH2SUA5xj-
dXgIThhJwFg34kStppn20EmskLdfgmS5gYYPAznrnXeZH7KW6fM1wqpfu7h3BJBriyK_qXWtIz-
JMR1d2YTy4QOw7_Y4dZPAQWcUCv-Qtpp-8pbUK7UXZzHu8p_Up7QMGuyNSzfvPxwzyB2teVlAh-
6ep7qm1PICTCPn1GTaPIwV_V9SaRRzL1hYM6K7o8zzHWWG3NtidRX279xK9snRyJFfH7IrhL9ag_eD2vdxC6lEJMaORKkh
bm9tDPgU6Fh0JLhc-
Bh8_NwManyZfIfdD_VzbzcNqZudNyygMC76JnPc2NVsznZKScihkrvnmvUQUw12DblDTAzcnzzf4qh_FdIXJqpXGYDKRaj
YucI6bIuP-toDyU2sKcyne--gzEhQ1Z_SNGYz_o7H8iZW2BTeE_N_dolcATFtxexGWY-
TfM3fvuNnO_pM6ZyvJe4xZYRs1wO82No8dK3dD6eyZI2LO6usWsfynFMaJFGazVX1pLHPOWB-
EvujJCyv_Qe9lagjR0Z26fLLc2jDiwOG8WT9ieVjCnEtlFGWkg54BoyQGUe31JNpSrmAyLxRyoqQRDA9IoJGxNZts9vGcW
RmVSQxpQfPHV-eIAxjACj2NUxL8N0zP4oBbBCAJ8","expires_in":3600,"token_type":"Bearer"}
```


3.3 Подключение POS-терминалов. Формирование Security Token с использованием приватного ключа, выданного УЦ банка

Терминал формирует токен безопасности (security token) и выполняет запрос токена доступа согласно разделу 6.3 стандарта RFC 7521 Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants. Для запроса используется токен безопасности в формате JSON Web Token (JWT) согласно разделу 2.1 стандарта JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants, утверждения которого заполняются следующим образом:

Утверждение (claim)	Обязательность, тип	Описание	Источник данных
Payload			
alg	да	Алгоритм шифрования.	

		Поддерживаются RS256, RS384, RS512 и эллиптические кривые на NIST P-256, NIST P-384, NIST P-512.	
iss	Да string	Issuer Для Security Token это официальный URL банка, либо сервиса СБП в банке Например: https://www.bank.ru/	Настройка терминала
sub	Да string	Имя системного пользователя/идентификатор терминала, доступен в Портале агента во вкладке «учетные записи».	Настройка терминала
aud	Да string	Audience Содержит адрес token endpoint Access Manager. Access Manager отклоняет запросы с HTTP-статус кодом 400, в случае если утверждение заполнено неправильно. Например: https://ips-merchant-dev.ekassir.com/am/ipslegals/connect/token	Настройка терминала
iat	Нет integer	Issued At Дата и время выпуска токена в UTC. Access Manager отклоняет запрос, если Issued At в будущем. Например, 1516239022.	Вычисляется
exp	Да integer	Expires At Дата и время срока действия токена в UTC. Access Manager должен отклонять запрос, если Expires At в прошлом. Например, 1516240022.	Вычисляется, срок жизни токена задается в настройках терминала (10-15 сек)
scope	Да string	Значение: *read*	
jti	Нет string	JWT ID Уникальный идентификатор токена. При наличии данного утверждения Access Manager выполнит проверку уникальности идентификатора и отклонит запрос, если проверка неуспешна.	GUID, генерится при каждом запросе токена
Signature			
Подпись	Да	Подпись JWT токена с использованием ключа длиной RSA 2048 байт и хэш-алгоритма SHA-256.	Вычисляется

		Для подписи токена используется приватный ключ выданный УЦ банка. Access Manager выполнит проверку подписи и отклонит запрос, если проверка неуспешна, с кодом ответа 400 и указанием в поле "error" кода ошибки "Token validation fault".	

Поля запроса Access Token заполняются следующим образом:

Поле	**Обязательность**	**Описание**	**Источник данных**
grant_type	Да	Всегда urn:ekassir.com:params:oauth:grant-type:pos-terminal	Константа
assertion	Да	Security Token в формате JWT	
scope	Да	Значение: *read*	
client_id	Да	Идентификатор приложения, для которого должен быть выдан доступ. Это - значение поля Application ID для Open ID Client, созданного для подключения кассового ПО. Open ID Client создается в realm системных пользователей на сервере AM (раздел Applications)-Подробно о настройке realm - в документе «Портал Агента ТСП. Администрирование» (раздел «Создание и настройка области безопасности для системных пользователей»).	Настройка терминала

3.4 Подключение POS-терминалов. Формирование Security Token с использованием приватного ключа, выданного УЦ производителя терминала

Терминал формирует токен безопасности (security token) и выполняет запрос токена доступа согласно разделу 6.3 стандарта RFC 7521 Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants. Отличие данного процесса от RFC в использовании нестандартного типа разрешения (grant_type). Для запроса используется токен безопасности в формате JSON Web Token (JWT) согласно разделу 2.1 стандарта JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants, утверждения которого заполняются следующим образом:

Утверждение (claim)	**Обязательность, тип**	**Описание**	**Источник данных**
alg	да	Алгоритм шифрования. Поддерживаются RS256, RS384, RS512 и эллиптические кривые на NIST P-256, NIST P-384, NIST P-512.	

x5c	Да string	X.509 сертификат открытого ключа, соответствующий приватному ключу, использованному для подписи Security Token. Должен содержать цепочку сертификатов, при этом сам сертификат открытого ключа должен быть первым сертификатом в массиве. Формат: base64-encoded в соответствии с RFC 7515	
Payload			
iss	Да string	Issuer Для Security Token это официальный URL банка, либо сервиса СБП в банке Например: https://www.bank.ru/	Настройка терминала
sub	Да string	Имя системного пользователя/идентификатор терминала, доступен в Портале агента во вкладке «учетные записи».	Настройка терминала
aud	Да string	Audience Содержит адрес token endpoint Access Manager. Access Manager должен отклонять запросы с HTTP-статус кодом 400, в случае если утверждение заполнено неправильно. Например: https://ips-merchant-dev.ekassir.com/am/ipslegals/connect/token	Настройка терминала
iat	Нет integer	Issued At Дата и время выпуска токена в UTC. Access Manager отклонит запрос, если Issued At в будущем. Например, 1516239022.	Вычисляется
exp	Да integer	Expires At Дата и время срока действия токена в UTC. Access Manager отклонит запрос, если Expires At в прошлом. Например, 1516240022.	Вычисляется, срок жизни токена задается в настройках терминала (10-15 сек)
scope	Да	Значение: *read*	
jti	Нет string	JWT ID Уникальный идентификатор токена. При наличии данного утверждения Access Manager выполнит проверку уникальности	GUID, генерится при каждом запросе токена

		идентификатора и отклонит запрос, если проверка неуспешна.	
Signature			
Подпись	Да	Подпись JWT токена с использованием ключа длиной RSA 2048 байт и хэш-алгоритма SHA-256. Для подписи токена используется приватный ключ выданный УЦ производителя терминалов. Access Manager должен выполнит проверку подписи и отклонит запрос, если проверка неуспешна, с кодом ответа 400 и указанием в поле "error" кода ошибки "Token validation fault".	Вычисляется

Поля запроса Access Token заполняются следующим образом:

Поле	Обязательность	Описание	Источник данных
grant_type	Да	Всегда urn:ekassir.com:params:oauth:grant-type:pos-terminal	Константа
assertion	Да	Security Token в формате JWT	
scope	Да	Значение: *read*	
client_id	Да	Идентификатор приложения, для которого должен быть выдан доступ. Это - значение поля Application ID для Open ID Client, созданного для подключения кассового ПО. Open ID Client создается в realm системных пользователей на сервере AM (раздел Applications)-Подробно о настройке realm - в документе «Портал Агента ТСП. Администрирование» (раздел «Создание и настройка области безопасности для системных пользователей»).	Настройка терминала

Пример claims:

```
Header:
{
alg:
"RS512"
x5c:
"MIIFIDCCAwigAwIBAgIJAI/CBlF4HjXVMA0GCSqGSIb3DQEBBQUAMBQxEjAQBgNVBAMTCXN1cGVyY2VydDAeFw0xNzA3M
TgxMzE3NDZaFw0yNzA5MDQxMzE3NDZaMBQxEjAQBgNVBAMTCXN1cGVyY2VydDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADC
CAgoCggIBAMxuMPL09eePnX7ev3e94UaA113hdh4PQhSQXTYR8aVDdq3r1JURKuaNm2F1rgjsSPpGXUaVr7w6l1PqrR2G0
mffxSSo7z6dfgDHietKF+W//MOtjDJiCzYyRbE8ByOptat8jSlJUKLCqurf5gDALx9Eo2qLxfRH5Iz/sFELpP2Rl1Jkg5W
1onwPG6MuW8zazGHMsb9WD2LSl9DR6gCCiRF8gOVu0M6u07NjpXt2wXl50vM7CmwMMEajU6ALaBacrb+0KL1XNxYuHy6jb
6p1CyPNDfQTN9+NMV1u19dYHw7rXeq+JnETGw8OQV/qbVJxtUEQbvDUSedt/N9jmPVuCJhUWSffcXaUMWjFDmU5Ej/Mqxk
```

```
l8SRvSdrwdkbWy7EoTMq5S/v7B3UuZanmZwBiAuEfAovyiBABEGfq5BaZftx80vXpgxW4iR/R5h86/IctT70osEcZUK6+o
Vk9xPUp6XBNbbjHi3m0mouFGdbmE0llfYXmXAh93WtNT+DqBWuNM5htTZ9Q3I1iLm4NDKzjs9HNIHdI3zy+C/Bt/WdAKh1
w3E/cc/SSmR7Hcrz0axS2kqx6URkKoFEmuWH1sK46TsfYSgQzx7DtXdthJLX8s1D/mbeVwByND7SqGZX+m8csn+cG/QtNp
7SdxHtpNKpXf7zhico7Z4ssCTncGKdu994BAgMBAAGjdTBzMB0GA1UdDgQWBBQm7wt1/HKVyfWcTMppFVhOzIGI6TBEBgN
VHSMEPTA7gBQm7wt1/HKVyfWcTMppFVhOzIGI6aEYpBYwFDESMBAGA1UEAxMJc3VwZXJjZXJ0ggkAj8IGUXgeNdUwDAYDV
R0TBAUwAwEB/zANBgkqhkiG9w0BAQUFAAOCAgEAFxJVPcTZP35Hv7tipSkYxyQi6qK/EeaSVmkzS+MWAZjnwYovsZkeUDI
BPa8T5yx2ZDKOBeNtwwp8k9Z7GRR+SQQNQWddAUOpM5t0ry+sglA4s+s3LzjPhGDdIjfrBzgzIIwq+i8jIjfanUIQLCBS4
grhkmK4h/Lsc2FeT4mz74RkF0+abcwWXoaM6wuKdkcVbsV/oB7AiLmBPmR/WAS+ntss1YigL7A2qXXpNYRimhTgRp9JXhH
09iInceJL9V5JvDJkF1JGr+eCpJ2oi4ryiyhdnPyKtncZuZEF2w/0GQ7ZOk2Po2XsN/Da7EtdqELD+AaKOlyIUIQ1Kx893
XGv9srPNxK2Oi1C5WrA48/gCJGpTARaYq7yhRijQTd/bMooAcFovdAnfM2xqvm5iet7N4dWTC7Uli2ZyHXbXspFCVvikv2
c6Bk4KJ15KbxTrkoNHjLUnO1tZ9Dqyq7BNDtwkaVkmetobnBqVtbWNJMWnlG12BEltXS4Yxn9SL8u3GOeyCU8cxJiLVjUj
p//BpOFPyqb/Zt/YIhv88+OFBwIRp1aatyV+CbnMV/r8WsrJ6X5ZEsqXtMpGPnqJ4ofokF+tqLL98jMpBjyA/tABWpLzym
tJ77Py14wmdGe1LKrwQmsWYSVqCgeRFqLFmheHnDZSP72h+ELfs39Hh+L6+5Z0ZM="
}

Payload:
{
  "aud": "https://machinename.com/am/ipslegals/connect/token",
  "iss": "https://www.bank.ru/",
  "sub": "SystUser5082",
  "scope": read,
  "exp": {{exp: 1602772272}},
  "iat": {{iat: 1602761472}},
  "jti": "{{jti: a34b568f6e1d4cf38ff9d1c8ed3803ca}}"
}
```


Запрос:


```
POST https://machinename.com /am/ipslegals/connect/token HTTP/1.1
Host: ips-merchant-dev.ekassir.com
Accept: application/json
X-Requested-With: XMLHttpRequest
X-Correlation-Id: CheckAuth
Request-Timeout: 30
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 3469

grant_type=urn%3Aekassir.com%3Aparams%3Aoauth%3Agrant-type%3Apos-
terminal&assertion=eyJhbGciOiJSUzUxMiIsIng1YyI6WyJNSUlGSURDQ0F3aWdBd0lCQWdJSkFJL0NCbEY0SGpYVk1
BMEdDU3FHU0liM0RRRUJCUVVBTUJReEVqQVFCZ05WQkFNVENYTjFjR1Z5WTJWeWREQWVGdzB4TnpBM01UZ3hNekUzTkRaY
UZ3MHlOekE1TURReE16RTNORFphTUJReEVqQVFCZ05WQkFNVENYTjFjR1Z5WTJWeWREQ0NBaUl3RFFZSktvWklodmNOQVF
FQkJRQURnZ0lQQURDQ0Fnb0NnZ0lCQU14dU1QTDA5ZWVQblg3ZXYzZTk0VWFBMTEzaGRoNFBRaFNRWFRZUjhhVkRkcTNyM
UpVUkt1YU5tMkYxcmdqc1NQcEdYVWFWcjd3NmwxUHFyUjJHMG1mZnhTU283ejZkZmdESGlldEtGK1cvL01PdGpESmlDell
5UmJFOEJ5T3B0YXQ4alNsSlVLTENxdXJmNWdEQUx4OUVvMnFMeGZSSDVJei9zRkVMcFAyUmwxSmtnNVcxb253UEc2TXVXO
HphekdITXNiOVdEMkxTbDlEUjZnQ0NpUkY4Z09WdTBNNnUwN05qcFh0MndYbDUwdk03Q213TU1FYWpVNkFMYUJhY3JiKzB
LTDFYTnhZdUh5NmpiNnAxQ3lQTkRmUVROOStOTVYxdTE5ZFlIdzdyWGVxK0puRVRHdzhPUVYvcWJWSnh0VUVRYnZEVVNlZ
HQvTjlqbVBWdUNKaFVXU2ZmY1hhVU1XakZEbVU1RWovTXF4a2w4U1J2U2Ryd2RrYld5N0VvVE1xNVMvdjdCM1V1WmFubVp
3QmlBdUVmQW92eWlCQUJFR2ZxNUJhWmZ0eDgwdlhwZ3hXNGlSL1I1aDg2L0ljdFQ3MG9zRWNaVUs2K29Wazl4UFVwNlhCT
mJiakhpM20wbW91RkdkYm1FMGxsZllYbVhBaDkzV3ROVCtEcUJXdU5NNWh0VFo5UTNJMWlMbTROREt6anM5SE5JSGRJM3p
5K0MvQnQvV2RBS2gxdzNFL2NjL1NTbVI3SGNyejBheFMya3F4NlVSa0tvRkVtdVdIMXNLNDZUc2ZZU2dReng3RHRYZHRoS
kxYOHMxRC9tYmVWd0J5TkQ3U3FHWlgrbThjc24rY0cvUXROcDdTZHhIdHBOS3BYZjd6aGljbzdaNHNzQ1RuY0dLZHU5OTR
CQWdNQkFBR2pkVEJ6TUIwR0ExVWREZ1FXQkJRbTd3dDEvSEtWeWZXY1RNcHBGVmhPeklHSTZUQkVCZ05WSFNNRVBUQTdnQ
lFtN3d0MS9IS1Z5ZldjVE1wcEZWaE96SUdJNmFFWXBCWXdGREVTTUJBR0ExVUVBeE1KYzNWd1pYSmpaWEowZ2drQWo4SUd
VWGdlTmRVd0RBWURWUjBUQkFVd0F3RUIvekFOQmdrcWhraUc5dzBCQVFVRkFBT0NBZ0VBRnhKVlBjVFpQMzVIdjd0aXBTa
1l4eVFpNnFLL0VlYVNWbWt6UytNV0Faam53WW92c1prZVVESUJQYThUNXl4MlpES09CZU50d3dwOGs5WjdHUlIrU1FRTlF
XZGRBVU9wTTV0MHJ5K3NnbEE0cytzM0x6alBoR0RkSWpmckJ6Z3pJSXdxK2k4aklqZmFuVUlRTENCUzRncmhrbUs0aC9Mc
2MyRmVUNG16NzRSa0YwK2FiY3dXWG9hTTZ3dUtka2NWYnNWL29CN0FpTG1CUG1SL1dBUytudHNzMVlpZ0w3QTJxWFhwTll
SaW1oVGdScDlKWGhIMDlpSW5jZUpMOVY1SnZESmtGMUpHcitlQ3BKMm9pNHJ5aXloZG5QeUt0bmNadVpFRjJ3LzBHUTdaT
2syUG8yWHNOL0RhN0V0ZHFFTEQrQWFLT2x5SVVJUTFLeDg5M1hHdjlzclBOeEsyT2kxQzVXckE0OC9nQ0pHcFRBUmFZcTd
5aFJpalFUZC9iTW9vQWNGb3ZkQW5mTTJ4cXZtNWlldDdONGRXVEM3VWxpMlp5SFhiWHNwRkNWdmlrdjJjNkJrNEtKMTVLY
nhUcmtvTkhqTFVuTzF0WjlEcXlxN0JORHR3a2FWa21ldG9ibkJxVnRiV05KTVdubEcxMkJFbHRYUzRZeG45U0w4dTNHT2V
5Q1U4Y3hKaUxWalVqcC8vQnBPRlB5cWIvWnQvWUlodjg4K09GQndJUnAxYWF0eVYrQ2JuTVYvcjhXc3JKNlg1WkVzcVh0T
XBHUG5xSjRvZm9rRit0cUxMOThqTXBCanlBL3RBQldwTHp5bXRKNzdQeTE0d21kR2UxTEtyd1Ftc1dZU1ZxQ2dlUkZxTEZ
taGVIbkRaU1A3MmgrRUxmczM5SGgrTDYrNVowWk09Il19.eyJpc3MiOiJodHRwczovL3d3dy5iYW5rLnJ1LyIsInN1YiI6
```

```
IlN5c3RVc2VyNTA4MiIsImF1ZCI6Imh0dHBzOi8vaXBzLW1lcmNoYW50LWRldi5la2Fzc2lyLmNvbS9hbS9pcHNsZWdhbH
MvY29ubmVjdC90b2tlbiIsImlhdCI6IjE2MDI3NjE0NzIiLCJleHAiOiIxNjAyNzcyMjcyIiwic2NvcGUiOiJyZWFkIiwi
anRpIjoiYTM0YjU2OGY2ZTFkNGNmMzhmZjlkMWM4ZWQzODAzY2EifQ.JzvEejMiZzrDrpNdOVBFQzH6V-
yZHK56yskEgb54LJQWWSbUZtbks-aiT18TZDe_t3qdIHzYXofGIkBF45NO8yt6zxTjl4fh0Frjbn-RRPEW-
bWfYOoTuSZo8LyBKBnSDFT0q1B8URymAfgP2br5MuCzAvrEGjRTHdk0YOIz9vtCt35M4qHMTs0oZbgJOjjhxBJhxZ8qPNY
NJHl4XgktMiyxJYq_LYHlCcilE9TPQJLryt5D0hirtmjytWsRSXtWBOnWKe3Iv2HPJ1rPrfQYlP4gzkGFvUMkP32_Y_kvS
cOdDKWcsBRoWzjO1zLVm8tYcUuQ2TEVBSC5x5D9n7CkvTP90l_GMWRVOE6vthgyY5F-
dEaE5c2RA3nmJm8ntJZAPLkegvw6-
3pGceXaOa2aR_NwmRfgsl46NaXqRo7Tn3gE0NMthNsKMtUHdlpZFue89LQumH3OEFH63vxp1zgpWQRnKPK0KDRbcoAJX3L
zj47TzPSkZWVSQZhN6vhW6AbV4lbY4Gsxd3OPmCopDfEczW-
DfCp7sbqzmeeBlZL4sT35JfMOgp1K4StDCdBSKrqcfilYmz6WAgyPukg8qufWIe5jeilRAaNsH-
kGr8wEaoBgfsL4CUWIT38nh09ghz5hbYf3AXh9YDnk3UDy3vmjcoxYKbD844CK5GHbfiA0vvg&client_id=pos-
terminal&scope=read
```

Ответ:

```
HTTP/1.1 200 OK
Server: nginx
Date: Thu, 15 Oct 2020 11:32:13 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Cache-Control: no-store, no-cache
Pragma: no-cache
Strict-Transport-Security: max-age=2592000
X-Correlation-Id: CheckAuth
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload

{"access_token":"eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1Q
zMxMzEiLCJ4NXQiOiJHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJuYmYiOjE2MDI3NjE1
MzMsImV4cCI6MTYwMjc2NTEzMywiaXNzIjoiaHR0cHM6Ly9pcHMtbWVyY2hhbnQtZGV2LmVrYXNzaXIuY29tL2FtL2lwc2
xlZ2FscyIsImF1ZCI6Imh0dHBzOi8vaXBzLW1lcmNoYW50LWRldi5la2Fzc2lyLmNvbS9hbS9pcHNsZWdhbHMvcmVzb3Vy
Y2VzIiwiY2xpZW50X2lkIjoicG9zLXRlcm1pbmFsIiwic2NvcGUiOiJyZWFkIiwic3ViIjoiU3lzdFVzZXI1MDgyIiwiYX
V0aF90aW1lIjoxNjAyNzYxNTMzLCJpZHAiOiJla2Fzc2lyLmFtIiwicm9sZSI6ImxlZ2FsX2FkbWluIiwibGVnYWxJZCI6
IkxBNDM4Njk5MjYzNyIsIm1lcmNoYW50SWQiOiJNQTcyNDg1Mjk1NDAiLCJhbXIiOlsidXJuOmVrYXNzaXIuY29tOnBhcm
FtczpvYXV0aDpncmFudC10eXBlOnBvcy10ZXJtaW5hbCJdfQ.e___Q-
98GJo72M8_TQphZUYNGRtIWUodWYd80VGzZuJEjfw0AJKd8pHbQ2h50F02DzSBrRkluLxld_tjAlNG6Ii53uExlQN_cWea
HU7FGGueSG9GwNID0dHpPPQI0W4UuJDRLS1AHeuQRPJOErlelyN04tug4J1nDB1UueqbcSctPpmFUXQosdqflYgTLm47gL
2PRKvswCz5fu3qdQAWJfysU14wrn1rXJ2g0V9gNUbK-cYB36GZl1DNzNMtFR_YY34Y-LyjCkPLFc8v-
OJnAaerFvBMcaPXGnLk7gYjDzhVc8yBjWz3d9jIK_as58zEGMXoiiX1mZEHMIpecevqikVFNy5F3qBmI7piDv7D7WAgObT
o5bjBKyya8vxTlhQaMHZK4SvCaSbBbY_6fpCmpdXQ_XNHxXLA-
r5EfNVvqZ1odd3ABzCODSCJ1zI3538exLSnk0KJgn5XmjyP2nyVO_71-
eETPUrzi3xwl5OSge2BVU7c8TqJpHgnPcauThLHj4qbzA1kptzb9XP9-
0E9lpVCast0B37MTPCsfFA5fvlNUncWzUpgxsQyL3CISdLQ0wjVTXDaYmq1H7KOWI4bBLd8Df0yydX17O3zBdJYnYZ2GqJ
7jeI7h_BDMBTdlAF3ff5M8FQigTABtJiQ9I9bnpm0ET-
4rbtg3CBKbWsG49A","expires_in":3600,"token_type":"Bearer"}
```

3.5 Устаревший вариант авторизации

Авторизация API осуществляется в соответствии со стандартом OAuth2 RFC 6749 (п 4.1) с помощью сервера Access Manager (AM). Для получения токена доступа используется Client Credentials grant type.

Для авторизации API необходимо выполнить запрос:

```
POST https://[AMbaseusrl]/[realm]/connect/token
```

где

AMbaseusrl – адрес AM

realm – название области безопасности, созданной в AM для юридических лиц (в примерах документации название области *ipsclients*) .

В теле запроса необходимо передать параметры :

```
grant_type=client_credentials&scope=read&client_id=[clientID]&client_secret=[clientSecret]
```

где

clientID – параметр *Application ID,* занесенный в АМ при создании OpenID Connect Client в области безопасности, созданной в АМ для юридических лиц.
clientSecret – параметр *Client Secret* для этого *Application ID*.

Пример запроса

```
POST https://ips-merchant-dev.ekassir.com/am/c2bclients/connect/token
200
1764 ms
Network
Request Headers
Content-Type: application/x-www-form-urlencoded
User-Agent: PostmanRuntime/7.26.1
Accept: */*
Cache-Control: no-cache
Postman-Token: 2161a393-7119-4afe-9da2-949bb267a729
Host: ips-merchant-dev.ekassir.com
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: 84
Request Body
grant_type=client_credentials&scope=read&client_id=LA0000000012&client_secret=qwe123
```

Пример ответа

При успешной обработке будет получен ответ, который будет содержать:

- токен доступа *access token*, тип токена *"token_type":"Bearer"*. Этот токен должен указываться в заголовке HTTP Authorization при выполнении запросов к API
- время жизни токена (параметр *expires_in*, время в секундах). Перед отправкой запроса требуется проверка, не истекло ли время жизни токена. В проверка случае отсутствия действующего токена необходимо выполнить запрос нового.

```
Response Header
Server: nginx/1.14.2
Date: Mon, 06 Jul 2020 15:01:02 GMT
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Cache-Control: no-store, no-cache
Pragma: no-cache
Strict-Transport-Security: max-age=2592000
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload
X-Correlation-Id: 0HM0PNJ71I2CJ:00000001
Content-Encoding: gzip
Response Body
{"access_token":"eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1Q
zMxMzEiLCJ4NXQiOiJHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJuYmYiOjE1OTQwNDc2
NjIsImV4cCI6MTU5NDA1MTI2MiwiaXNzIjoiaHR0cHM6Ly9pcHMtbWVyY2hhbnQtZGV2LmVrYXNzaXIuY29tL2FtL2MyYm
NsaWVudHMiLCJhdWQiOiJodHRwczovL2lwcy1tZXJjaGFudC1kZXYuZWthc3Npci5jb20vYW0vYzJiY2xpZW50cy9yZXNv
dXJjZXMiLCJjbGllbnRfaWQiOiJMQTAwMDAwMDAwMTIiLCJzY29wZSI6InJlYWQifQ.EaMX2yFER7uNaeyTUqGcpsNAtcF
RDfoKyq3pyp4gbkn0SRZYZHnqDGnX2SY7cKldr8L-1pReNKgFjL15JGMqqtgL4HJgjigDWJe-
_vvrh92iXy6AIP3Tet7vLgEoJxIA1Mp1OeGGmOBmdEj2_NKLEuSYkM7trU0Ss_qhEUTVEn8P2x0U9uhgdRBhwPeZ4PwuhP
zOGM4zaY9_goH3yMw3pEi4zTSf9N2Ews0HcW92Pk3ZJE194B6bO4GW91FqAXDCRUqL0NCfHZLoA7FnWuwIk3Vjz1GMyhnU
wpUvdqbpAzfS2oGZLJwG3r1klEJHXgLZdCxmyCBHS19cgSf1oc6l8O83j3bC5PaL8YZ9mUNuykqUY96amb8aW4cj87sPuF
gIkTBoS6Kck5iRKMX66d_W8hwWOKkVT2nwT9efqMDAnQ0YanXrboQGGjcstE4-0ubNHrENBLT6hdor1LhzpEFwLlZYaK-
KtIstX3BhkYQuKlDVUqX1xXAVy7ZDZtZr5cYLDobo8Xe3w4djxcEyEe0FC8Clo7Lwdh7htTuTKGyj2-
b2tbmlfBA9vyLpaa47Ccc7hyUo-
```

```
aroOnleP9X9X6OblWImAjGiFusjWFdqAxJ6DWYgxxRMAVzCZMFaNLqD1sXUKmNEGlD1n6VgqB2n9iiVPybXUTph-
dPfLnt3KV7O0h8","expires_in":3600,"token_type":"Bearer"}
```


4 Точка входа

Для обработки операций (работа с QR-кодами, реестрами платежей, инициирование возврата) используются методы с точкой входа:

`https://[baseurl]/api/merchant/v1`

где *baseurl* – базовый адрес API, определяется при установке (см документ *Merchant API Service.Администрирование*).

Для получения справочника участников СБП используются методы с точкой входа:

`https://[baseurl]/api/members/v1`

где *baseurl* – базовый адрес API, определяется при установке (см документ *Merchant API Service.Администрирование*).

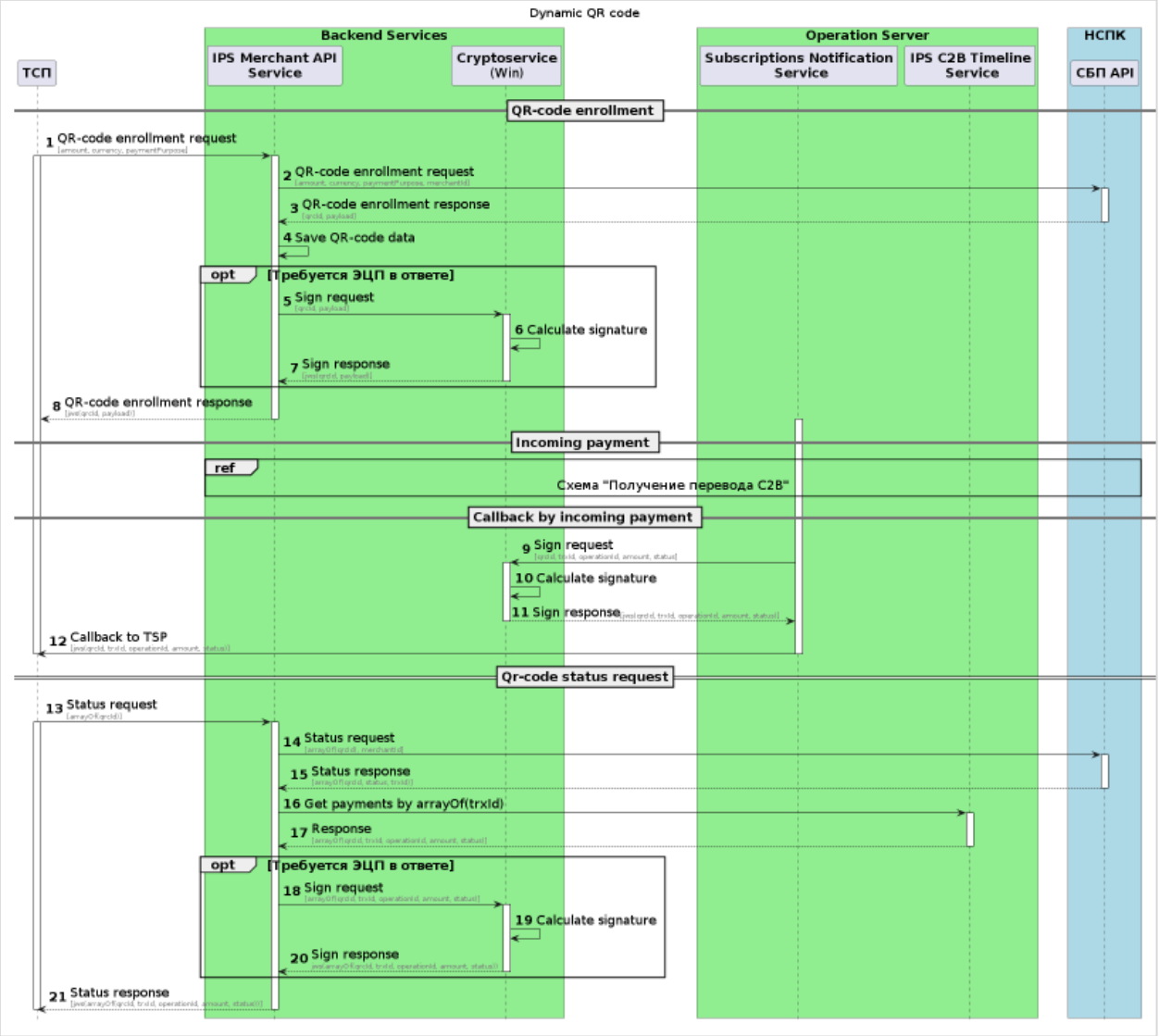
5 Регистрация и получение QR-кодов

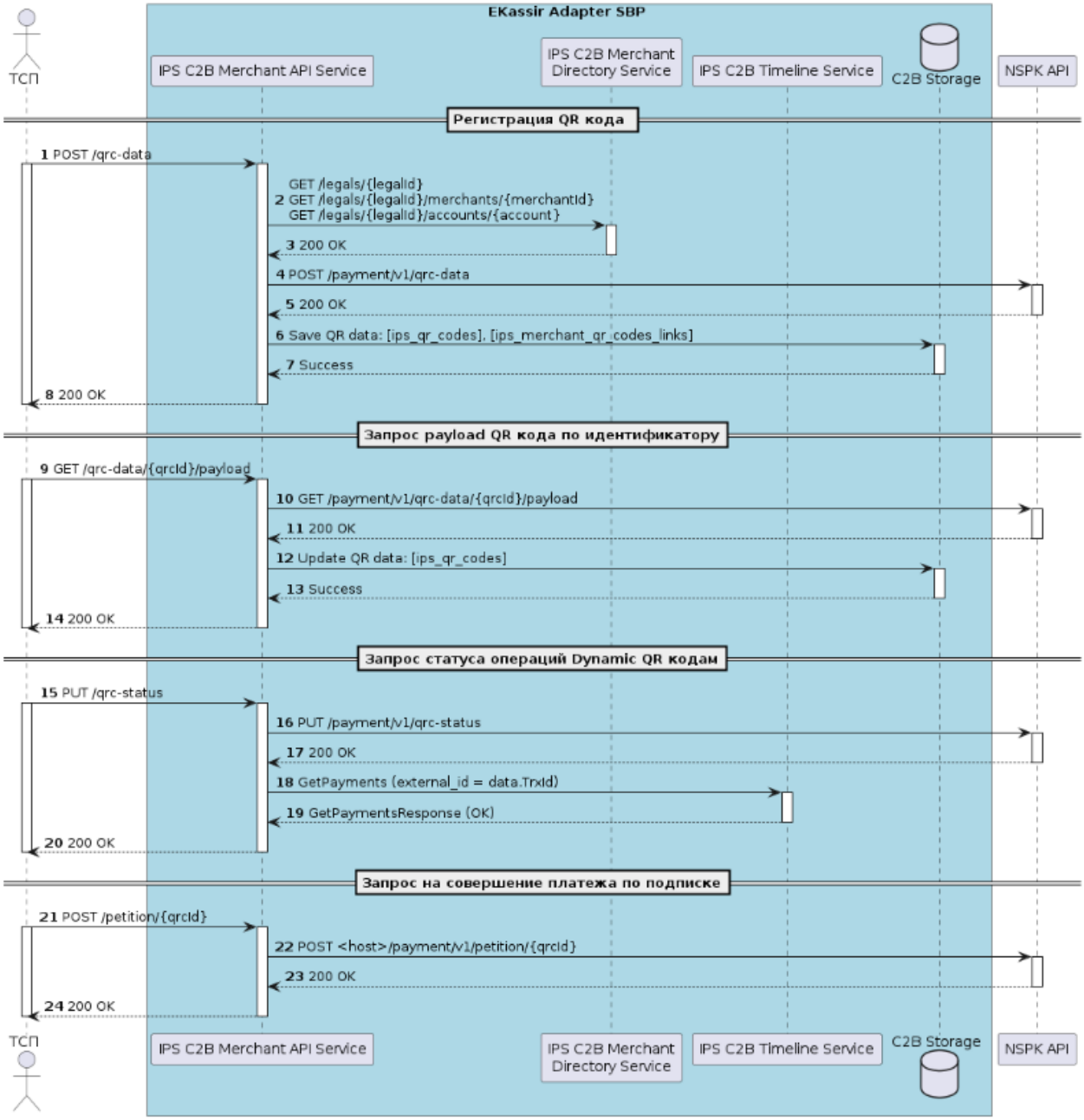
Методы Merchant API для работы с QR-кодами предназначены для:

- Регистрации QR-кода в СБП, при этом возвращается payload кода (URL с параметрами), опционально можно запросить получение изображения QR-кода.
- Запроса payload QR-кода по идентификатору зарегистрированного кода.
- Запроса статуса операции по Dynamic QR-кодам.
- Выполнение привязки счета с немедленной оплатой или без оплаты

Полное описание API находится в файлах *Merchant-API.yaml* , в формате OpenAPI 3.0.

5.1 Схема работы с динамическими QR-кодами





5.2 Формирование сообщения с содержимым application/jose

Для получения ответа, подписанного ЭЦП, вызов API должен производиться с использованием заголовка

Accept: application/jose

Ответное сообщение формируется в формате JSON Web Signature (JWS) в соответствии с с RFC 7515 (п. 7.1 *JWS Compact Serialization*):

Тело сообщения формируется из трех составляющих следующим образом:

(Base64Url(Header) || '.' || Base64Url(Payload) || '.' || Base64Url(Signature)).
где

Header - JSON-объект, содержащий алгоритм подписи и UUID сертификата клиента, парный приватному ключу, которым подписан запрос.
Payload - JSON-документ с реквизитным составом сообщения.
Signature - подпись данных приватной частью ключевой пары клиента (используется приватный ключ парный сертификату клиента с UUID, указанному в Заголовке (Header) в параметре kid). Подпись вычисляется по алгоритму указанному в Заголовке (Header) в параметре alg и вычисляется от исходных данных: Base64Url(Header) || '.' || Base64Url(Payload).

Пример ответа в формате JWS на запрос QR-кода (тело ответа)

```
ewogICJhbGciOiAiZ29zdDM0LjEwLTIwMTIiLAogICJraWQiOiAiYjk5NDM4MWQtZDg4Ni00YjVhLTk0MDctMTY3Mjk1MjYxMGFhIgp9.ewoicXJjSWQiOiAiQVMxMDAwNjcwTFNTN0ROMThTSlFETlA0QjA1S0xKTDIiLAoicGF5bG9hZCI6ICJodHRwczovL3FyLm5zcGsucnUvQVMxMDAwNjcwTFNTN0ROMThTSlFETlA0QjA1S0xKTDI_dHlwZT0wMSZiYW5rPTEwMDAwMDAwMDAwMSZzdW09MTAwMDAmY3VyPVJVQiZjcmM9QzA4QiIKfQ==.SNM2l3cktEuhlcbvxM49v3Mk9vNtgvQfG_JR4bHhDc5dDZMwDkmNdesZ9TGHk7lFQHp_zoTH8KBSDbb-PmTn6g
```

5.3 Получение QR-кода

Для получения QR-кода используется POST-запрос с точкой входа:

`POST /qrc-data`

Кроме payload кода, который будет получен в ответе, с помощью query-параметров можно запросить получение изображения кода.

Query-параметры запроса:

- *mediaType*– формат изображения, возможные значения: *image/png, image/svg+xml*
- *width* – ширина изображения (по умолчанию 300)
- *height* – высота изображения (по умолчанию 300)

Замечание. Под учетной записью ТСП может быть сгенерирован QR-код только для данного ТСП. При генерации QR-кода значение утверждения *merchantId* токена доступа (при наличии) должно совпадать со значением идентификатора ТСП в теле запроса. При несовпадении, сервис вернет ответ с HTTP-статусом 403.

Пример запроса

Тело запроса

```
{"account":"40702810100010000001",
"40702810100010000001",
"MerchantId":"MA0000000011",
"TemplateVersion":"01",
"QrcType":"02",
"Amount":1000,
"Currency":"RUB",
"QrcTtl: 60"
"PaymentPurpose":"покупка",
"params": {
"paymentData": " покупка "
 }
}
```

Параметры запроса

Параметр	Обязательный	Описание
account	нет	Номер счета юридического лица
merchantId	да	Идентификатор ТСП

templateVersion	да	Версия payload QR-кода (01 – Версия 1)
qrcType	да	Тип QR-кода • 01 - QR-Static (QR наклейка) • 02 - QR-Dynamic (QR на кассе). Используется для единоразовых оплат, в сценариях привязки счета используется для оплаты по привязанному счету, оплаты с привязкой счета. • 03 - QR-Subscription (QR для оформления привязки счета)
amount	нет	Сумма в копейках
qrcTtl	нет	Срок жизни Функциональной ссылки СБП в минутах. Необязательное поле. Минимальное допустимое значение - "1". Максимальное допустимое значение - "129 600" (90 дней). Внимание! Рекомендация ОПКЦ СБП: устанавливать срок жизни Функциональной ссылки СБП не менее "5" минут. Если поле "qrTtl" не передано в запросе, будет использовано значение по умолчанию - "4 320" минут (3 суток). Неприменимо для qrcType = "01".
currency	нет	Валюта операции
subscriptionPurpose	нет	Информация от ТСП, детализирующая цель привязки счета. Поле обязательно для qrcType = "03" (QR-Subscription) и для qrcType = "02" в сценарии "Оплата с привязкой счета"
paymentPurpose	нет	Дополнительная информация от ТСП
redirectUrl	нет	Содержит ссылку для автоматического возврата Плательщика из приложения Банка в приложение или на сайт ТСП. Допускаются только символы в кодировке ASCII. Формат должен соответствовать правилам кодировки URL.
Params - произвольные параметры		
paymentData	нет	Служебное поле назначения платежа (не передается в НСПК)

Пример ответа

В ответе получен код 201 – Операция успешно выполнена.


```
POST /api/merchant/v1/qrc-data; Elapsed: 00:00:03.3241148; StatusCode: 201; ContentLenght: 160
{"qrcId":"AD4E53BB8A3F4F2DA3C22929A168B730","payload":"https://qr.nspk.ru/AD4E53BB8A3F4F2DA3C2
2929A168B730?type=01&bank=500000000001&sum=1000&cur=RUB&crc=42M6"}
```


Параметры ответа

Параметр	Обязательный	Описание
qrcId	да	Идентификатор зарегистрированного QR кода в СБП
Payload	да	Payload кода – URL с параметрами (*qr.nspk.ru/идентификатор зарегистрированного QR-кода?Тип QR-кода&Идентификатор Банка Получателя (ТСП) (Member_ID&)Сумма в копейках&Валюта операции&Значение контрольной суммы*)
Image - изображение QR кода		

mediaType	нет	Формат получаемого файла: image/png - png, image/svg+xml - xml of svg Enum: [image/png, image/svg+xml]
content	нет	base64encoded image. Формат для декодирования зависит от mediaType

5.3.1 Callbacks запроса

5.3.1.1 Получение платежа

При получении платежа по идентификатору QR-кода Merchant API формирует задачу на отправку уведомления в ТСП на предварительно зарегистрированный адрес.

Алгоритм работы

При получении уведомления о входящем платеже по ранее зарегистрированному QR-коду происходит обращение к сервису *CryptoService (Windows)* для получения электронной подписи и затем отправка POST-запроса на web-адрес ТСП:

> **`POST notification_base_url`** с заголовком **Content-Type: application/jose**
>
> где *notification_base_url* - предварительно зарегистрированный адрес для отправки уведомлений во внешнюю систему.

Уведомление имеет формат JSON Web Signature (JWS) в соответствии с RFC 7515 (п. 7.1 *JWS Compact Serialization*):

Payload содержит следующие поля:

Параметр	Обязательный	Описание
qrcId	да	Идентификатор зарегистрированного QR кода в СБП
trxId	да	Идентификатор Операции в ОПКЦ СБП
paramsId	нет	Идентификатор активных значений параметров Платежной ссылки СБП. Передается, если оплата происходила по Кассовой ссылке СБП. Параметр передается в Subscriptions Notification Service (см. документ "eKassir Адаптер для СБП C2B").
amount	да	Сумма операции в копейках
operationId	нет	Номер операции в платежной системе банка, возвращается для успешной операции
operationTimestamp	да	Дата и время операции в UTC, возвращается для успешной операции
status	да	Статус операции (completed - ACWP или rejected – RJCT)
paymentData	нет	Служебное поле назначения платежа
parameters	нет	Произвольные параметры

Пример десериализованного Payload:

```
{
  "qrcId": "649634e3a6974351a2d480ef21069e85",
  "trxId": "X1A2S3D5F6G7H8J9K0C4S5C6D7V5D1K2"
  "amount": 19900
  "operationId": 2456532
  "status": "completed"
  "operationTimestamp  "2020-01-31T08:55:24.225Z"
  "paymentData": "покупка"
}
```

В случае успешно найденного платежа ТСП возвращает ответ с кодом 200.

5.3.1.2 Уведомление о привязке счета

При получении уведомления о результате оформления привязки счета по идентификатору QR-кода происходит отправка уведомления в ТСП на предварительно зарегистрированный адрес.

Алгоритм работы

При получении уведомления о результате оформления привязки счета происходит обращение к сервису *CryptoService (Windows)* для получения электронной подписи и затем отправка POST-запроса на web-адрес ТСП:

> **POST notification_base_url/subscription/notification** с заголовком **Content-Type: application/jose**
> где *notification_base_url* - предварительно зарегистрированный адрес для отправки уведомлений во внешнюю систему.

Уведомление имеет формат JSON Web Signature (JWS) в соответствии с RFC 7515 (п. 7.1 *JWS Compact Serialization*):

Payload содержит следующие поля:

Параметр	Обязательный	Описание
code	да	Код ответа оформления привязки счета
message	да	Описание кода ответа
status	да	Статус оформления привязки счета
qrcId	да	Идентификатор зарегистрированного QR кода в СБП
subscriptionToken	нет	Идентификатор привязки счета, назначаемый Банком Плательщика
memberId	нет	Идентификатор Банка Плательщика

Пример десериализованного Payload:

```
{
  "code": "RS00000",
  "message": "Запрос обработан успешно",
  "status": "ACWP",
  "qrcId": "AD1000038AQ1FG409KGACUQU5QRSH3FM",
  "subscriptionToken": "70LSS7DN18SJQRS10006DNPKLJL24B05"
}
```

В случае успешно найденной привязки ТСП возвращает ответ с кодом 200.

5.3.1.3 Уведомление о финальном статусе

Уведомление ТСП о финальном статусе Операции СБП С2В от ОПКЦ СБП:

> **POST notification_base_url/qrc/notification** с заголовком **Content-Type: application/jose**
> где *notification_base_url* - предварительно зарегистрированный адрес для отправки уведомлений во внешнюю систему.

Уведомление имеет формат JSON Web Signature (JWS) в соответствии с RFC 7515 (п. 7.1 *JWS Compact Serialization*):

Payload содержит следующие поля:

Параметр	Обязательный	Описание
qrcId	да	Идентификатор зарегистрированной Платежной ссылки СБП
trxId	да	Идентификатор Операции СБП С2В
paramsId	нет	Идентификатор активных значений параметров Платежной ссылки СБП. Передается, если оплата происходила по Кассовой ссылке СБП
status	да	Статус оформления привязки счета
amount	да	Сумма Операции СБП С2В в копейках. Целое, положительное число.
errorCode	нет	Зарезервированное поле для будущего использования
message	нет	Описание кода ответа
timestamp	да	Дата и время выполнения Операции СБП С2В
payerId	да	Маскированный номер телефона Клиента-Плательщика
kzo	нет	Контрольное Значение Операции СБП. Формируется ОПКЦ СБП на основании собственных алгоритмов. Присутствует при условии, что Операция СБП С2В по запрашиваемому qrcId завершена успешно (параметр "status" ="ACWP").

Пример десериализованного Payload:

```
{
  "qrcId": "AD1000038AQ1FG409KGACUQU5QRSH3FM",
  "trxId": "A10560039016760100000046815985938",
  "paramsId": "837462903765830298564738290187 35",
  "status": "ACWP",
  "amount": 5000,
  "errorCode": "string",
  "message": "string",
  "timestamp": "2021-03-02T17:15:01.148",
  "payerId": "*********6731",
  "kzo":
"FDOS4JUETLYT639ADAFZ4GAUY9VSM2TG2Y595LQ20EKQF3JM1CIV4ZTZYA1EYIMFMEJSRB2UR7KATMA29QYMY9KLA2NHR
HZ1SKEJ947DTTS67M1GZ6DDUXGJWUXH0Z7X"
}
```

В случае успеха возвращает ответ с кодом 200.

5.4 Запрос payload QR кода по идентификатору

Для получения payload QR кода используется GET-запрос с точкой входа:

`GET /qrc-data/{qrcId}/payload`

где *qrcId* - идентификатор зарегистрированного QR кода в СБП. Обязательный параметр

Query-параметры запроса:

- *mediaType*– формат изображения, возможные значения: *image/png, image/svg+xml*
- *width* – ширина изображения (по умолчанию 300)
- *height* – высота изображения (по умолчанию 300)

Пример запроса

```
GET /api/merchant/v1/qrc-data/AD4E53BB8A3F4F2DA3C22929A168B730/payload
```

Параметры запроса

Параметр	Описание
AD4E53BB8A3F4F2DA3C22929A168B730	qrcId - идентификатор зарегистрированного QR кода в СБП

Пример ответа

В ответе получен код 200 – успешный ответ.

```
GET /api/merchant/v1/qrc-data/AD4E53BB8A3F4F2DA3C22929A168B730/payload; Elapsed:
00:00:02.8348812; StatusCode: 200; ContentLenght: 160
{"qrcId":"AD4E53BB8A3F4F2DA3C22929A168B730","payload":"https://qr.nspk.ru/AD4E53BB8A3F4F2DA3C2
2929A168B730?type=01&bank=500000000001&sum=1000&cur=RUB&crc=42M6"}
```

Параметры ответа

Параметр	Обязательный	Описание
qrcId	да	Идентификатор зарегистрированного QR кода в СБП
payload	да	Payload кода – URL с параметрами (*qr.nspk.ru/идентификатор зарегистрированного QR-кода?Тип QR-кода&Идентификатор Банка Получателя (ТСП) (Member_ID&)Сумма в копейках&Валюта операции&Значение контрольной суммы*)
Image - изображение QR кода		
mediaType	нет	Формат получаемого файла: image/png - png, image/svg+xml - xml of svg Enum: [image/png, image/svg+xml]
content	нет	base64encoded image. Формат для декодирования зависит от mediaType

5.5 Запрос статуса операций по Dynamic QR кодам

Для получения статуса операций по QR кодам используется PUT-запрос с точкой входа:

`PUT /qrc-status`

Пример запроса

В теле запроса перечисляются идентификаторы QR-кодов:

```
{"qrcIds": ["AD4E53BB8A3F4F2DA3C22929A168B730"]}
```

Параметры запроса

Параметр	Описание
qrcIds	Список кодов, по которым необходимо получить статусы операций.

Примеры ответа

Код ответа 200 «Успешный ответ»

Тело ответа:

```
{
  "data": [
    {
      "qrcId": "AS10005FPRU69C21926AB7GK2HSSOP8N",
      "code": "RQ00000",
      "message": "Запрос обработан успешно",
      "status": "ACWP",
      "operationId": 100000123,
      "trxId": "X1A2S3D5F6G7H8J9K0C4S5C6D7V5D1K2",
      "amount": 100000,
      "operationTimestamp": "2020-01-31T08:55:24.225Z",
      "params": {
        "paymentData": "string"
      },
      "paymentParams": [
        {
          "name": "parameter_name",
          "value": "parameter_value"
        }
      ]
    }
  ]
}
```

Параметры ответа

Параметр	Обязательный	Описание
data		Массив статусов операций по QR кодам
qrcId	да	Идентификатор зарегистрированного QR кода в СБП
code	да	Код ответа на запрос статуса QR кода (RQ00000)
message	да	Описание кода ответа на запрос (Запрос обработан успешно)
status	нет	Статус операции, инициированной Dynamic QR кодом (NTST - NOT_STARTED операции по QR коду не существует)
operationId	нет	Номер операции в платежной системе банка, возвращается для успешной операции
trxId	нет	Идентификатор операции, инициированной Dynamic QR кодом

amount	нет	Сумма операции в копейках
operationTimestamp	нет	Дата и время операции в UTC, возвращается для успешной операции
Params - произвольные параметры		
paymentData		Служебное поле назначения платежа (не передается в НСПК)
paymentParams - дополнительные параметры платежа		
name		Наименование параметра
value		Значение параметра

5.6 Запрос на выполнение платежа по привязанному счету

Для запроса на выполнение платежа по привязанному счёту используется POST-запрос с точкой входа:

`POST /petition/{qrcId}`

где *qrcId* - идентификатор зарегистрированного QR-кода в СБП.

Пример запроса

Тело запроса

```
{
  "subscriptionToken": "70LSS7DN18SJQRS10006DNPKLJL24B05",
  "memberId": "100000000001"
}
```

Параметры запроса

Параметр	Обязательный	Описание
subscriptionToken	да	Токен, полученный при оформлении привязки счёта.
memberId	да	Идентификатор Банка Плательщика

Пример ответа

В ответе получен код 200 – Успешный ответ

```
{
  "code": "RQ00000",
  "message": "Запрос обработан успешно"
}
```

Параметры ответа

Параметр	Обязательный	Описание
code	да	Код ответа
message	да	Описание кода ответа

5.6.1 Callback запроса

5.6.1.1 Уведомление о разрешении на совершение платежа

При получении уведомления о разрешении на совершение платежа по привязанному счёту Merchant API формирует задачу на отправку уведомления в ТСП на предварительно зарегистрированный адрес.

Алгоритм работы

При получении уведомления о разрешении на совершение платежа по привязанному счёту происходит обращение к сервису *CryptoService (Windows)* для получения электронной подписи и затем отправка POST-запроса на web-адрес ТСП:

> **`POST notification_base_url/notification`** с заголовком **Content-Type: application/jose**
>
> где *notification_base_url* - предварительно зарегистрированный адрес для отправки уведомлений во внешнюю систему.

Уведомление имеет формат JSON Web Signature (JWS) в соответствии с RFC 7515 (п. 7.1 *JWS Compact Serialization*):

Payload содержит следующие поля:

Параметр	**Описание**
code	Код ответа оформления привязки счета
message	Описание кода ответа
qrcId	Идентификатор зарегистрированного QR кода в СБП

Пример десериализованного Payload:

```
{
  "code": "RQ00030",
  "message": "Согласие на проведение платежа",
  "qrcId": "AD1000038AQ1FG409KGACUQU5QRSH3FM"
}
```

6 Работа с реестрами платежей

Методы Merchant API для работы формирования реестров получателей по входящим C2B платежам предназначены для:

- Создания заявки для генерации реестра платежей (запросы от получателей).
- Запроса статуса готовности отчета по идентификатору заявки

Схема последовательности вызовов API



1. ТСП вызывает метод API для создания заявки на генерацию реестра платежей. В качестве параметров передаются дата платежей и тип реестра.
2. Ставится задача для получения реестра, ТСП получает в ответ идентификатор созданной заявки.
3. ТСП производит вызов API для получения статуса по созданной заявке. В качестве параметра передается идентификатор заявки, полученный на предыдущем шаге.
4. ТСП получает в ответ статус *created* и *url* для получения реестра (если реестр уже сгенерирован).
5. ТСП запрашивает реестр по полученному *url*.
6. ТСП получает архив, состоящий из реестра и файла с подписью.
7. Альтернативно: ТСП получает в ответ статус *processing* (если реестр еще НЕ сгенерирован)
8. Альтернативно: ТСП получает в ответ статус *error* (если произошла ошибка получения реестра)

6.1 Создание заявки для генерации реестра платежей

Для создания заявки для генерации реестра платежей используется POST-запрос с точкой входа:

POST /reports/orders

Возможно выгружать реестры только по определенным ТСП и за определенный период времени.

В параметре merchantId перечисляются идентификаторы ТСП, платежи которых должны попасть в выгрузку. При отсутствии этого параметра в запросе реестр выгружается по всем ТСП.

В запросе необходимо указывать либо пару StartDate+EndDate, либо параметр ReportDate.

Пример запроса

Тело запроса:

```
{
  "reportType": "TSCA",
  "merchants": [
    "MF0000000001"
  ],
  "startDate": "2020-01-31T08:55:24.225Z",
  "endDate": "2020-01-31T08:55:24.225Z"
}
```

Параметры запроса

Параметр	Описание
reportType	Тип отчета (TSCA - реестр успешных входящих платежей)
merchants	Перечислить идентификаторы ТСП, платежи которых должны попасть в выгрузку.
startDate	Дата и время начала периода. Если время или часовой пояс не указаны, то платежи выгружаются на указанную дату по времени Мск
endDate	Дата и время окончания периода. Если время или часовой пояс не указаны, то платежи выгружаются на указанную дату по времени Мск
ReportDate	Дата платежей, входящих в реестр.

Пример ответа

В ответе получен код 201 – успешный ответ.

```
POST /api/merchant/v1/reports/orders; Elapsed: 00:00:03.8933638; StatusCode: 201;
ContentLenght: 50
{"orderId":"ac9395d5-9a07-4191-8a67-ec3cd05d10eb"}
```

Параметры ответа

Параметр	Описание
orderId	Идентификатор заявки для получения реестра платежей.

6.2 Запрос статуса готовности отчета по идентификатору заявки

Для запроса статуса готовности отчета используется GET-запрос с точкой входа:

GET /reports/orders/{orderId}

orderId - Идентификатор заявки для получения реестра платежей.

Для получения реестра необходимо загрузить файл с URL, указанного в ответе.

Пример запроса

Тело запроса:

```
GET /api/merchant/v1/reports/orders/ac9395d5-9a07-4191-8a67-ec3cd05d10eb
```

Пример ответа - реестр в обработке

В ответе получен код 201 – успешный ответ.

```
GET /api/merchant/v1/reports/orders/ac9395d5-9a07-4191-8a67-ec3cd05d10eb; Elapsed:
00:00:00.0116246; StatusCode: 200; ContentLenght: 23"
{"status":"processing"}
```

Пример ответа - ссылка на реестр

```
{"status":"created","url":"https://ips-merchant-
dev.ekassir.com/reports/0137227aa24649608e6c89fb7a3a7ff1.zip"
```

Параметры ответа

Параметр	Описание
status	Статус генерации реестра платежей (created – реестр создан).
url	URL-адрес для получения реестра платежей. Отсюда необходимо загрузить файл с реестром. **Внимание. Для загрузки файла с реестром необходимо отправить авторизованный запрос на полученный URL. В запрос необходимо добавить заголовок Authorization с токеном доступа *access_token*, полученным при авторизации API (см. раздел Авторизация API).**

7 Инициирование возврата платежей

Методы для инициирования возврата платежей предназначены для:

- Проверки возможности осуществления операции возврата.
- Выполнения операции.
- Проверка статуса операции.

Для обработки операций используются методы POST, PUT и GET с точкой входа:

`https://[baseurl]/api/merchant/v1/operations/return`

Полное описание API находится в файле *B2C-Operation-Service.yaml* в формате OpenAPI 3.0.

7.1 Проверка. POST-запрос

Проверка возможности осуществления операции возврата, получение данных по клиенту: ФИО и номер счета.

`POST /operations/return`

При передаче параметра *recipientBankId* в теле запроса создание операции осуществляется в 2 запроса (POST - проверка возможности, PUT - выполнение операции)

1. Операция проводится в один этап (не передается параметр *recipientBankId*)

Пример запроса

```
POST /api/merchant/v1/operations/return
Body:
{  "ReferenceId":95291
   "id":"531700424b0a43589cc3d7e105abfbf3",
}
```

Параметры запроса

Параметр	Описание
ReferenceId	Референсный идентификатор исходной Операции СБП.
id	Уникальный идентификатор операции

Пример ответа

В ответе получен код 201 «Операция завершена, финальный статус операции необходимо получить из поля status.»

```
POST /api/merchant/v1/operations/return; Elapsed: 00:00:18.1512656; StatusCode: 201;
ContentLenght: 369
{"operationId":95295,"status":{"id":"completed","name":"Завершена"},
"session":"531700424b0a43589cc3d7e105abfbf3","pam":"Петр Петрович П",
"externalId":"1964c5f5759648818e2c2c6b609500bb","id":"531700424b0a43589cc3d7e105abfbf3",
"phone":"70000000000","amount":{"amount":10000,"currencyCode":"RUB"},
"recipientBankId":"500000000001","referenceId":95293}
```

Параметры ответа

Параметр	Описание
operationId	Номер операции в платежной системе банка
status.id	Идентификатор статуса (completed – завершена успешно)
status.name	Наименование статуса
session	Сессия, в рамках которой выполняется взаимодействие.
pam	Получатель
externalId	Идентификатор операции в СБП
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках
amount.currencyCode	Валюта операции
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API). Если идентификатор явно не указан в запросе то будет использоваться идентификатор банка отправителя C2B платежа.
referenceId	Референсный идентификатор исходной Операции СБП.

2. Запрос на возможность проведения операции, параметр *recipientBankId* передается.

Операция не будет проведена без последующего PUT запроса.

Пример запроса

```
POST /api/merchant/v1/operations/return
Body: {"Id":"c72353ba-536e-4f69-9a8a-e705d6137f16",
```

```
"Phone":"70000000000",
"Amount":{"Amount":100,"CurrencyCode":"RUB"},
"RecipientBankId":"500000000001",
"Purpose":"возврат",
"ReferenceId":95309}
```


Параметры запроса

Параметр	Описание
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках
amount.currencyCode	Валюта операции
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API). Если идентификатор явно не указан в запросе, то будет использоваться идентификатор банка отправителя C2B платежа.
purpose	Назначение платежа
ReferenceId	Референсный идентификатор исходной Операции СБП.

Пример ответа

В ответе получен код 200 «Возможно зачислить сумму перевода на счет Получателя.»


```
POST /api/merchant/v1/operations; Elapsed: 00:00:00.8225121; StatusCode: 200; ContentLenght:
324
{"session":"2fb0179e23f64b48ae5a91656fdc1386106006","pam":"Петр Петрович
П","externalId":"0d76cc01eb424650bcaafe7ddb2e6823","id":"c72353ba536e4f699a8ae705d6137f16","ph
one":"70000000000","amount":{"amount":100,"currencyCode":"RUB"},"recipientBankId":"50000000000
1","purpose":"возврат","referenceId":95309}
```


Параметры ответа

Параметр	Описание
session	Сессия, в рамках которой выполняется взаимодействие.
pam	Получатель
externalId	Идентификатор операции в СБП
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API). Если идентификатор явно не указан в запросе то будет использоваться идентификатор банка отправителя C2B платежа.
purpose	Назначение платежа
ReferenceId	Референсный идентификатор исходной Операции СБП.

7.2 Выполнение. PUT-запрос

Выполнение операции – возврат. Метод используется для возврата, при переданном *recipientBankId* в POST запросе.

`PUT /operations/return/{id}`

id – Идентификатор операции

Пример запроса

```
PUT /api/merchant/v1/operations/return/a91f71da12594cdd9c81963a15f88d54
Body:
{"Id":"a91f71da12594cdd9c81963a15f88d54","Phone":"70000000000","Amount":{"Amount":100,"Currenc
yCode":"RUB"},"RecipientBankId":"500000000001","Purpose":"возврат","ReferenceId":95311,"Sessio
n":"8f599d69a8ae478fa84e57143cca8438106007","Pam":"Петр Петрович
П","ExternalId":"52ea7c0c98bd4304a77029f8895ac312"}
```

Параметры запроса

Параметр	Описание
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API)..
purpose	Назначение платежа
ReferenceId	Референсный идентификатор исходной Операции СБП.
session	Сессия, в рамках которой выполняется взаимодействие.
pam	Получатель
externalId	Идентификатор операции в СБП

Пример ответа

В ответе передан код 201 «Операция завершена, финальный статус операции необходимо получить из поля status».

```
PUT /api/merchant/v1/operations/return/a91f71da12594cdd9c81963a15f88d54; Elapsed:
00:00:00.6607597; StatusCode: 201; ContentLenght: 394
{"operationId":95312,"status":{"id":"completed","name":"Завершена"},"session":"a91f71da12594cd
d9c81963a15f88d54","pam":"Петр Петрович
П","externalId":"52ea7c0c98bd4304a77029f8895ac312","id":"a91f71da12594cdd9c81963a15f88d54","ph
one":"70000000000","amount":{"amount":100,"currencyCode":"RUB"},"recipientBankId":"50000000000
1","purpose":"возврат","referenceId":95311}
```

Параметры ответа

Параметр	Описание
operationId	Номер операции в платежной системе банка
status.id	Идентификатор статуса (completed – завершена успешно)
status.name	Наименование статуса
session	Сессия, в рамках которой выполняется взаимодействие.
pam	Получатель
externalId	Идентификатор операции в СБП
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках

amount.currencyCode	Валюта операции
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API). Если идентификатор явно не указан в запросе то будет использоваться идентификатор банка отправителя C2B платежа.
purpose	Назначение платежа
referenceId	Референсный идентификатор исходной Операции СБП.

7.3 Проверка. GET-запрос

Проверка статуса операции. Метод используется для получения статуса операции, если он не был получен ранее.

`GET/operations/return/{id}`

id – Идентификатор операции

Пример запроса

```
GET /api/merchant/v1/operations/return/a1b74f75fed6411e8ee82a5e5c2ad0b1
```

Пример ответа

В ответе передан код 200 «Данные операции получены».

```
GET /api/merchant/v1/operations/a1b74f75fed6411e8ee82a5e5c2ad0b1; Elapsed: 00:00:00.2103235;
StatusCode: 200;
ContentLenght: 369
{"operationId":95318,"status":{"id":"completed","name":"Завершена"},
"session":"a1b74f75fed6411e8ee82a5e5c2ad0b1",
"pam":"Петр Петрович П","externalId":"761c2c73fd874a45959ad7bbceb78ae8",
"id":"a1b74f75fed6411e8ee82a5e5c2ad0b1",
"phone":"70000000000","amount":{"amount":10000,"currencyCode":"RUB"},"recipientBankId":"500000
000001","referenceId":95317}
```

Параметры ответа

Параметр	Описание
operationId	Номер операции в платежной системе банка
status.id	Идентификатор статуса (completed – завершена успешно)
status.name	Наименование статуса
session	Сессия, в рамках которой выполняется взаимодействие.
pam	Получатель
externalId	Идентификатор операции в СБП
id	Уникальный идентификатор операции
phone	Телефон получателя
amount.amount	Сумма в копейках
amount.currencyCode	Валюта операции
recipientBankId	Идентификатор Банка Получателя из справочника участников СБП (Members-API).
referenceId	Референсный идентификатор исходной Операции СБП.

8 Методы для получения справочника участников СБП

Для получения справочника используются методы с точкой входа:

`https://[baseurl]/api/members/v1`

8.1 Получение списка участников СБП. GET- запрос

`GET /`

Query-параметры запроса:

- *query* – Содержит значение, по которому необходимо произвести поиск в справочнике. Значение: *name*- название участника СБП
- *expand* – Дополнить объект дополнительными данными. Значение: *logo* - логотип банка
- *scenario* - Сценарий, который должен поддерживать участник СБП. Scenario не ограничен существующими на данный момент сценариями операций СБП, список может быть расширен.
  Возможные значения:
  - C2CPush - Быстрый перевод денег между физическими лицами C2C Push
  - Me2MePush - Быстрый перевод денег самому себе Me2Me Push
  - Me2MePull - Запрос перевода со своих счетов самому себе Me2Me Pull
  - C2BQRD - QR на кассе
  - C2BQRS - QR-наклейка
  - C2BMcom - Мгновенный счет
  - B2COther - Прочие выплаты физическому лицу
  - C2BRfnd - Возврат по Операции СБП C2B
  - Другие новые сценарии операций СБП

  Для указания нескольких сценариев необходимо использовать следующие операции:
  - and - логическое И
  - or - логическое ИЛИ Условия разделены между собой символом пробела - " "

  Для испоьзования фильтра по роли требуется после сценария добавить символ "/" и указать одно из следующих значений:
  - Sender - только роль sender
  - Receiver - только роль receiver
  - All - должны присутствовать обе роли

Полное описание API находится в файле *Members-API.yaml* в формате OpenAPI 3.0.

Примеры запроса

1. Запрос списка всех участников

```
GET https://ips-merchant-dev.ekassir.com/api/members/v1
Request Headers
X-Correlation-ID: TST426
Content-Type: application/json
```

```
Authorization: Bearer
eyJhbGciOiJSUzI1NiIsImtpZCI6IjE5RTY3Q0VCMjM3N0Y2MzMzMURDNTE3MjY0NDBCQkZFM0E1QzMxMzEiLCJ4NXQiOi
JHZVo4NnlOMzlqTXgzRkZ5WkVDN19qcGNNVEUiLCJ0eXAiOiJKV1QifQ.eyJuYmYiOjE1OTgxOTUzMDAsImV4
Host: ips-merchant-dev.ekassir.com
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: 64
```

2. Запрос участника по названию

```
GET https://ips-merchant-dev.ekassir.com/api/members/v1?query= Золотой Банк
```

3. Запрос участника по поддерживаемому сценарию

```
https://ips-merchant-dev.ekassir.com/api/members/v1?scenario=C2BQRS
```

Пример с несколькими сценариями:

```
https://ips-merchant-dev.ekassir.com/api/members/v1?scenario=C2BQRS/Sender and (C2CPush/Sender
or B2COther/All)
```

4. Запрос списка участников с логотипами

```
GET https://ips-merchant-dev.ekassir.com/api/members/v1?expand=logo
```

Примеры ответа

1. Запрос списка всех участников. Фрагмент тела ответа:


```
{
	"items": [
		{
			"id": "188888888885",
			"displayName": "Золотой Банк",
			"bic": "044525416",
			"scenarios": {
				"C2CPush": [
					"Sender",
					"Receiver"
				],
				"Me2MePush": [
					"Sender",
					"Receiver"
				],
				"Me2MePull": [
					"Sender",
					"Receiver"
				],
				"C2BQRD": [
					"Sender",
					"Receiver"
				],
				"C2BQRS": [
					"Sender",
					"Receiver"
				],
				"C2BMcom": [
					"Sender"
				],
				"B2COthr": [],
				"C2BRfnd": [
					"Sender",
					"Receiver"
				]
			}
```

```
		},
		{
			"id": "690000000005",
			"displayName": "Компас Плюс",
			"bic": "044525723",
			"scenarios": {
				"C2CPush": [
					"Sender",
					"Receiver"
				],
				"Me2MePush": [
					"Sender",
					"Receiver"
				],
				"Me2MePull": [],
				"C2BQRD": [
					"Sender",
					"Receiver"
				],
				"C2BQRS": [
					"Sender",
					"Receiver"
				],
				"C2BMcom": [
					"Sender"
				],
				"B2COthr": [],
				"C2BRfnd": []
			}
		},
		{
			"id": "110000000099",
			"displayName": "Модульбанк_2",
			"bic": "043469751",
			"scenarios": {
				"C2CPush": [
					"Sender",
					"Receiver"
				],
				"Me2MePush": [
					"Sender",
					"Receiver"
				],
				"Me2MePull": [],
				"C2BQRD": [],
				"C2BQRS": [],
				"C2BMcom": [],
				"B2COthr": [],
				"C2BRfnd": []
			}
		},}
```


2. Запрос участника по названию. Фрагмент тела ответа:


```
"items": [
		{
			"id": "188888888885",
			"displayName": "Золотой Банк",
			"bic": "044525416",
			"scenarios": {
				"C2CPush": [
					"Sender",
					"Receiver"
				],
				"Me2MePush": [
					"Sender",
					"Receiver"
```

```
                        ],
                        "Me2MePull": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BQRD": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BQRS": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BMcom": [
                                "Sender"
                        ],
                        "B2COthr": [],
                        "C2BRfnd": [
                                "Sender",
                                "Receiver"
                        ]
                }
        },
```

3. Запрос участника по поддерживаемому сценарию. Фрагмент тела ответа:

```
"items": [
        {
                "id": "188888888885",
                "displayName": "Золотой Банк",
                "bic": "044525416",
                "scenarios": {
                        "C2CPush": [
                                "Sender",
                                "Receiver"
                        ],
                        "Me2MePush": [
                                "Sender",
                                "Receiver"
                        ],
                        "Me2MePull": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BQRD": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BQRS": [
                                "Sender",
                                "Receiver"
                        ],
                        "C2BMcom": [
                                "Sender"
                        ],
                        "B2COthr": [],
                        "C2BRfnd": [
                                "Sender",
                                "Receiver"
                        ]
                }
        },
        {
                "id": "690000000005",
                "displayName": "Компас Плюс",
```

```
                        "bic": "044525723",
                        "scenarios": {
                                "C2CPush": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "Me2MePush": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "Me2MePull": [],
                                "C2BQRD": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "C2BQRS": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "C2BMcom": [
                                        "Sender"
                                ],
                                "B2COthr": [],
                                "C2BRfnd": []
                        }
                },
                {
                        "id": "100000000201",
                        "displayName": "Банк Кремлевский",
                        "bic": "044525196",
                        "scenarios": {
                                "C2CPush": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "Me2MePush": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "Me2MePull": [],
                                "C2BQRD": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "C2BQRS": [
                                        "Sender",
                                        "Receiver"
                                ],
                                "C2BMcom": [
                                        "Sender"
                                ],
                                "B2COthr": [],
                                "C2BRfnd": []
                        }
                },
                {
```


4. Запрос списка участников с логотипами. Фрагмент тела ответа:


```
{
        "items": [
                {
                        "id": "188888888885",
                        "displayName": "Золотой Банк",
                        "logo": {
```

```
                        "mediaType": "image/png",
                        "data":
iVBORw0KGgoAAAANSUhEUgAABEoAAARKCAIAAADXJAK3AAAABmJLR0QA/wD...YII="
      },
                    "bic": "044525416",
                    "scenarios": {
                           "C2CPush": [
                                  "Sender",
                                  "Receiver"
                           ],
                           "Me2MePush": [
                                  "Sender",
                                  "Receiver"
                           ],
                           "Me2MePull": [
                                  "Sender",
                                  "Receiver"
                           ],
```

Параметры ответа

Параметр	Описание
id	Идентификатор участника СБП
displayName	Наименование участника СБП
Logo	Объект для передачи изображения
mediaType	Формат передаваемого файла: • image/png - png • image/svg+xml - xml of svg Enum: Array [2]
data	Формат для декодирования (зависит от mediaType)
bic	Банковский идентификатор участника СБП
scenarios	Список сценариев
Элементы списка *Название сценария: [* "*Sender*", "*Receiver*" *],*	Сценарии, поддерживаемые участником СБП, с указанием доступных ролей: Отправитель/Получатель. Возможные сценарии: o C2CPush - быстрый перевод денег между физическими лицами o Me2MePush - быстрый перевод денег самому себе o Me2MePull - запрос перевода со своих счетов самому себе o C2BQRD - QR на кассе o C2BQRS - QR-наклейка o C2BMcom - мгновенный счет o B2COthr - прочие выплаты физическому лицу o C2BRfnd - возврат по операции СБП С2 Scenario не ограничен существующими на данный момент сценариями операций СБП, поэтому в ответ может прийти сценарий, не указанный в списке выше. Например, `"C2CPush": [ "Sender",`

	"Receiver"],

8.2 Получение данных участника СБП по идентификатору. GET-запрос

GET /{memberId}

Пример запроса

```
GET https://ips-merchant-dev.ekassir.com/api/members/v1/188888888885
```

Пример ответа

```
{
    "id": "188888888885",
    "displayName": "Золотой Банк",
    "bic": "044525416",
    "scenarios": {
        "C2CPush": [
            "Sender",
            "Receiver"
        ],
        "Me2MePush": [
            "Sender",
            "Receiver"
        ],
        "Me2MePull": [
            "Sender",
            "Receiver"
        ],
        "C2BQRD": [
            "Sender",
            "Receiver"
        ],
        "C2BQRS": [
            "Sender",
            "Receiver"
        ],
        "C2BMcom": [
            "Sender"
        ],
        "B2COthr": [],
        "C2BRfnd": [
            "Sender",
            "Receiver"
        ]
    }
}
```

Параметры ответа

Параметр	Описание
id	Идентификатор участника СБП
displayName	Наименование участника СБП
bic	Банковский идентификатор участника СБП
scenarios	Список сценариев
Элементы списка *Название сценария: [*	Сценарии, поддерживаемые участником СБП, с указанием доступных ролей: Отправитель/Получатель.

"*Sender*", "*Receiver*" *],*	Возможные сценарии: o C2CPush - быстрый перевод денег между физическими лицами o Me2MePush - быстрый перевод денег самому себе o Me2MePull - запрос перевода со своих счетов самому себе o C2BQRD - QR на кассе o C2BQRS - QR-наклейка o C2BMcom - мгновенный счет o B2COthr - прочие выплаты физическому лицу o C2BRfnd - возврат по операции СБП С2 Scenario не ограничен существующими на данный момент сценариями операций СБП, поэтому в ответ может прийти сценарий, не указанный в списке выше. Например, `"C2CPush": [ "Sender", "Receiver" ],`

8.3 Получение логотипа участника СБП в виде файла. GET-запрос

`GET /{memberId}/logo`

Пример запроса

```
GET https://ips-merchant-dev.ekassir.com/api/members/v1/500000000001/logo
```

Пример ответа

В ответе передается **Пример ответа**

логотип :



9 Методы для работы с платежными ссылками

API для работы с платежными ссылками.

СБП позволяет проводить оплату с использованием Кассовой ссылки СБП. Это позволит многократно использовать неизменяемый QR-код, переопределяя значения части параметров Платежной ссылки СБП.

ТСП однократно регистрирует платежную ссылку. Далее, для проведения оплаты, активирует ссылку, передавая соответствующие параметры для оплаты. Если покупатель передумал совершать покупку, то Кассовая ссылка должна быть деактивирована:

- Деактивирована может быть только активированная Кассовая ссылка СБП.
- Кассовая ссылка СБП не может быть деактивирована, если начата процедура оплаты.

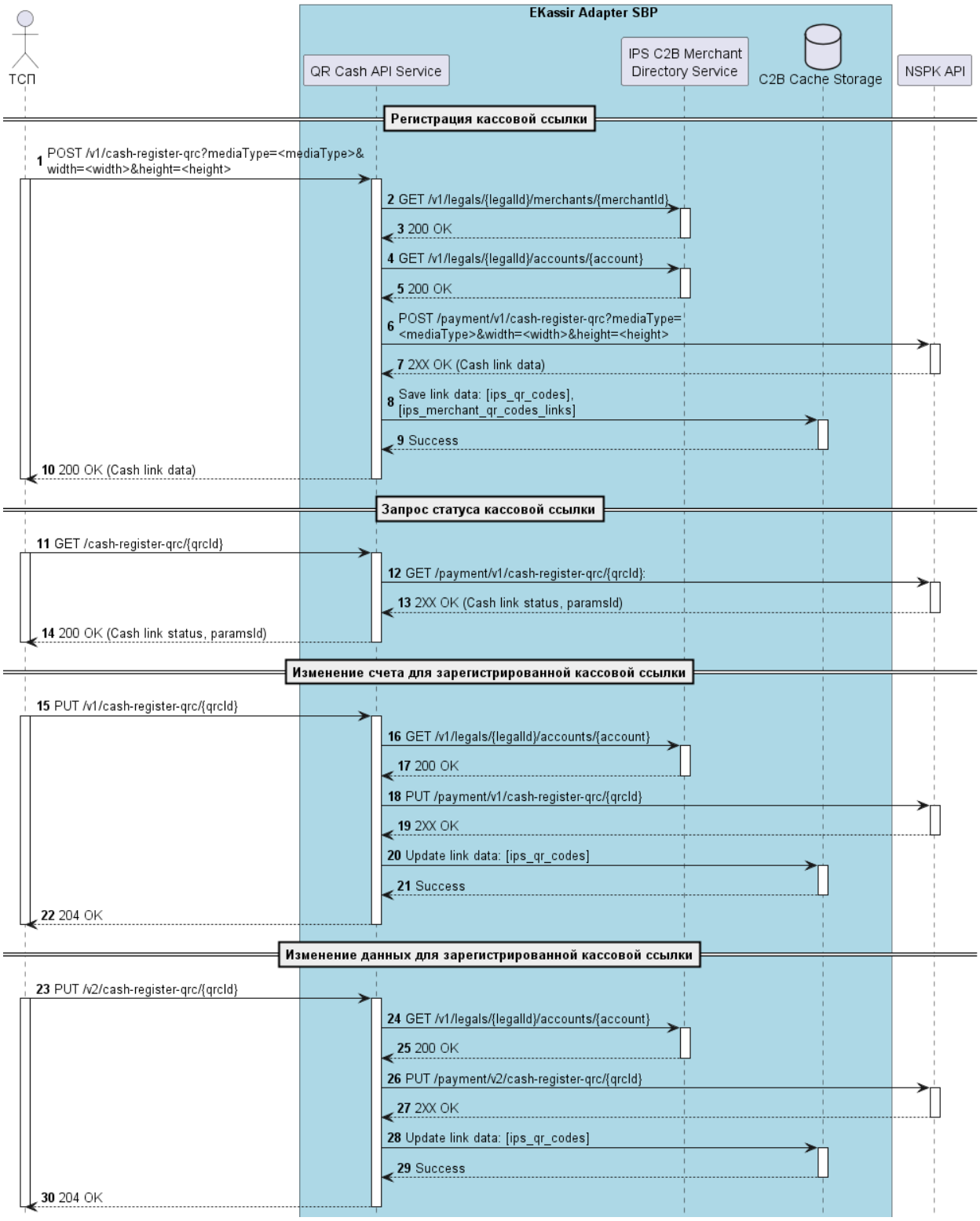
Если ТСП необходимо изменить счёт в зарегистрированной кассовой ссылке, необходимо воспользоваться методом изменения платежной ссылки. Смена счета юридического лица для зарегистрированной Кассовой ссылки СБП возможна только для деактивированной Кассовой ссылки СБП.

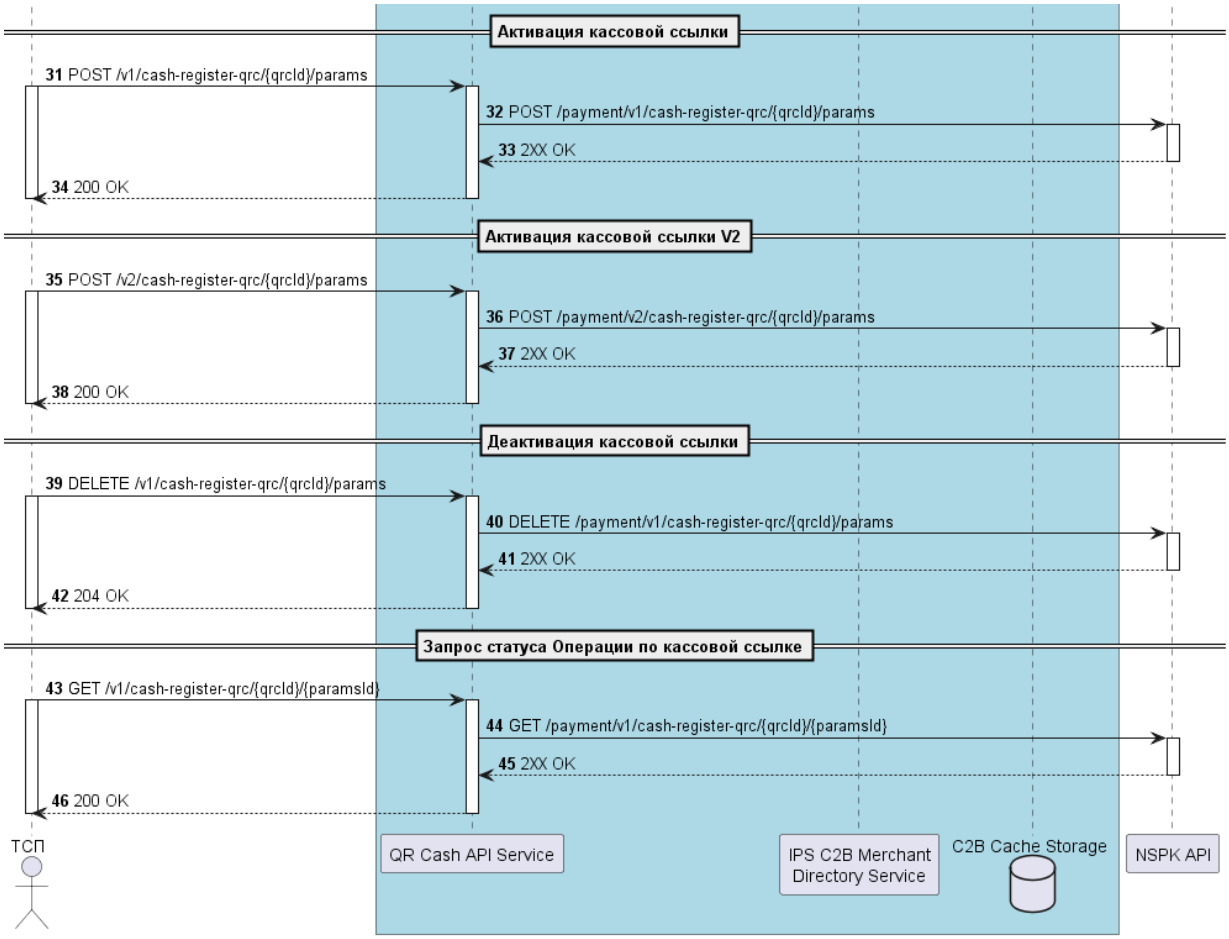
Оплата по Кассовой ссылке СБП – стандартная операция C2B, которая учитывается в Транзакционных отчетах наравне с другими транзакциями C2B. Узнать статус входящей операции C2B по кассовой ссылке можно с помощью получения callback или выполнения запроса qrc-status.

Методы API для работы с Кассовыми ссылками:

- Регистрация платежной ссылки
- Изменение данных платежной ссылки
- Запрос статуса Кассовой ссылки
- Активация платежной ссылки
- Деактивация платежной ссылки
- Статус Операции по Кассовой ссылке

Схема последовательности вызовов API





9.1 Регистрация платежной ссылки. POST – запрос

Потоки 1, 10.

POST/v1/cash-register-qrc

Query-параметры:

- *mediaType* - Mime-type изображения QR кода (image/png или image/svg+xml)
- *width* – ширина изображения, значение по умолчанию: *300*
- *height* – высота изображения, значение по умолчанию: *300*

При выполнении запроса может осуществляется проверка на принадлежность идентификатора ТСП из запроса к ЮЛ, которое получало токен. Проверка будет выполняться в случае, если в токене указан идентификатор ТСП. При несоответствии в ответе будет передана ошибка "*OQ05005: "ТСП с указанным идентификатором не найдено*". При соответствии данные о счете юрлица могут быть использованы для дальнейшей проверки счета (см. ниже).

Также выполняется проверка статуса счета юридического лица. Используется номер счета из запроса, если в запросе не указан номер счета, то используется номер, полученный в результате проверки принадлежности ТСП выше. Если значение получить не удалось или указан некорректный счет, то в ответе будет ошибка "*OQ05004: Счет юридического лица не*

найден". Если будет возвращен статус blockall, blocking или closed, то в ответе будет ошибка "*OQ05011: Счет юридического лица заблокирован или закрыт*".

Пример запроса

```
{
  "account": "40702810100010000001",
  "merchantId": "MF0000000001",
  "redirectUrl": "https://exampletsp.io/qwertyui"
}
```

Параметры запроса

Параметр	Описание
account	Счет ЮЛ, ИП или самозанятого. Необязательный параметр.
merchantId	Идентификатор ТСП. Обязательный параметр.
redirectUrl	Содержит ссылку для автоматического возврата Плательщика из приложения Банка в приложение или на сайт ТСП. Формат должен соответствовать правилам кодировки URL.

Пример ответа

Код ответа 200 – «Платежная ссылка успешно зарегистрирована»

```
{
  "qrcId": "AS100055MOHT58Q18CCBQGUOU5DQDIPR",
  "payload": "string",
  "image": {
    "mediaType": "image/png",
    "content": "iVBORw0KGgoAAAANSUhEUgAABEoAAARKCAIAAADXJAK3AAAABmJLR0QA/wD...YII="
  }
}
```

Параметры ответа

Параметр	Описание
qrcId*	Идентификатор зарегистрированного QR кода в СБП
payload*	Payload зарегистрированного QR кода в СБП
image	
mediaType*	Формат получаемого файла: image/png - png, image/svg+xml - xml of svg Enum: [image/png, image/svg+xml]
content*	Формат для декодирования зависит от mediaType. base64 encoded image.

9.2 Запрос статуса Кассовой ссылки. GET – запрос

Потоки 11, 14.

GET/v1/cash-register-qrc/{qrcId}

{qrcId} - идентификатор зарегистрированного QR кода в СБП.

Пример ответа

```
{
  "status": "WAITING_PAYMENT",
  "paramsId": "837462903765830298564738290187 35"
}
```

Параметры ответа

Параметр	Описание
status	Статус Кассовой ссылки • "INACTIVATED" - Ссылка не активирована • "WAITING_PAYMENT" - Кассовая ссылка СБП активирована и готова к оплате • "IN_PROGRESS" - Операция по Кассовой ссылке СБП в процессе выполнения
paramsId*	Идентификатор активных значений параметров Платежной ссылки СБП, которые передаются Агентом ТСП в запросе на активацию

9.3 Изменение счета для зарегистрированной кассовой ссылки. PUT – запрос

Потоки 15, 22.

PUT/v1/cash-register-qrc/{qrcId}

{qrcId} - идентификатор зарегистрированного QR кода в СБП.

При выполнении запроса также происходит проверка статуса счета и его принадлежность к ЮЛ. Если статус счета равен blockall, blocking или closed, то будет получен код 422 «Счет юридического лица заблокирован или закрыт»

Если счет не соответствует юр. лицу или мерчанту, то будет получен код 422 «Счет юридического лица не найден»

Пример запроса

```
{
  "account": "40702810100010000001"
}
```

Параметры запроса

Параметр	Описание
account*	Счет ЮЛ, ИП или самозанятого

Пример ответа

Код ответа 204 «Запрос выполнен успешно».

9.4 Изменение данных для зарегистрированной Кассовой ссылки. PUT – запрос

Потоки 23, 30.

PUT/v2/cash-register-qrc/{qrcId}

{qrcId} - идентификатор зарегистрированного QR кода в СБП.

При выполнении запроса также происходит проверка статуса счета и его принадлежность к ЮЛ. Если статус счета равен blockall, blocking или closed, то будет получен код 422 «Счет юридического лица заблокирован или закрыт»

Если счет не соответствует юр. лицу или мерчанту, то будет получен код 422 «Счет юридического лица не найден»

Пример запроса

```
{
  "account": "40702810100010000001",
  "merchantId": "MF0000000001",
  "redirectUrl": "https://exampletsp.io/qwertyui"
}
```

Параметры запроса

Параметр	Описание
account*	Счет ЮЛ, ИП или самозанятого
merchantId	Идентификатор ТСП
redirectUrl	Содержит ссылку для автоматического возврата Плательщика из приложения Банка в приложение или на сайт ТСП. Допускаются только символы в кодировке ASCII. Формат должен соответствовать правилам кодировки URL.

Пример ответа

Код ответа 204 «Запрос выполнен успешно».

9.5 Активация Кассовой ссылки. POST – запрос

Потоки 31, 34.

POST/v1/cash-register-qrc/{qrcId}/params

{qrcId} - идентификатор зарегистрированного QR кода в СБП.

Пример запроса

```
{
  "amount": 10000,
  "paymentPurpose": "Капучино 300 мл 1 шт., булочка 1 шт.",
  "fraudScore": "0000000000000000",
  "qrcTtl": 10,
  "currency": "RUB"
}
```

Параметры запроса

Параметр	Описание
amount*	Сумма в копейках
paymentPurpose	Назначение платежа
fraudScore	Индикатор Подозрительной Операции Агента ТСП
qrcTtl	Период использования Кассовой ссылки в минутах. Минимальное значение 5 минут. Максимальное значение 20 минут. Если поле не передано в запросе, период использования по умолчанию составит - 5 минут.
currency	Валюта операции: RUB - Российский рубль Enum: Array [1]

Пример ответа

```
{
  "paramsId": "8.374629037658304e+31",
```

```
  "qrcId": "AS100055MOHT58Q18CCBQGUOU5DQDIPR",
  "amount": 10000,
  "currency": "RUB",
  "paymentPurpose": "Капучино 300 мл 1 шт., булочка 1 шт."
}
```

Параметры ответа

Параметр	Описание
paramsId*	Идентификатор активных значений параметров Платежной ссылки СБП, которые передаются Агентом ТСП в запросе на активацию
qrcId*	Идентификатор зарегистрированного QR кода в СБП
amount*	Сумма в копейках
currency	Валюта операции: RUB - Российский рубль Enum: Array [1]
paymentPurpose	Назначение платежа

9.6 Активация Кассовой ссылки V2. POST – запрос

Потоки 35, 38.

POST/v2/cash-register-qrc/{qrcId}/params

{qrcId} - идентификатор зарегистрированного QR кода в СБП.

Пример запроса

```
{
  "amount": 10000,
  "paymentPurpose": "Капучино 300 мл 1 шт., булочка 1 шт.",
  "fraudScore": "0000000000000000",
  "qrcTtl": 10,
  "currency": "RUB",
  "redirectUrl": "https://exampletsp.io/qwertyui"
}
```

Параметры запроса

Параметр	Описание
amount*	Сумма в копейках
paymentPurpose	Назначение платежа
fraudScore	Индикатор Подозрительной Операции Агента ТСП
qrcTtl	Период использования Кассовой ссылки в минутах. Минимальное значение 5 минут. Максимальное значение 20 минут. Если поле не передано в запросе, период использования по умолчанию составит - 5 минут.
currency	Валюта операции: RUB - Российский рубль Enum: Array [1]

redirectUrl	Содержит ссылку для автоматического возврата Плательщика из приложения Банка в приложение или на сайт ТСП. Допускаются только символы в кодировке ASCII. Формат должен соответствовать правилам кодировки URL.

Пример ответа

```
{
  "paramsId": "83746290376583029856473829018735",
  "qrcId": "AS100055MOHT58Q18CCBQGUOU5DQDIPR",
  "amount": 10000,
  "currency": "RUB",
  "paymentPurpose": "Капучино 300 мл 1 шт., булочка 1 шт.",
  "fraudScore": "0000000000000000",
  "redirectUrl": "https://exampletsp.io/qwertyui"
}
```

Параметры ответа

Параметр	Описание
paramsId*	Идентификатор активных значений параметров Платежной ссылки СБП, которые передаются Агентом ТСП в запросе на активацию
qrcId*	Идентификатор зарегистрированного QR кода в СБП
amount*	Сумма в копейках
currency	Валюта операции: RUB - Российский рубль Enum: Array [1]
paymentPurpose	Назначение платежа
fraudScore	Индикатор Подозрительной Операции Агента ТСП
redirectUrl	Содержит ссылку для автоматического возврата Плательщика из приложения Банка в приложение или на сайт ТСП. Допускаются только символы в кодировке ASCII. Формат должен соответствовать правилам кодировки URL.

9.7 Деактивация Кассовой ссылки. DELETE- запрос

Потоки 39, 42.

DELETE /cash-register-qrc/{qrcId}/params

{qrcId} - идентификатор зарегистрированного QR кода в СБП

Статус активации при получении запроса на деактивацию от ТСП не проверяется.

Пример ответа

Код 204 «Запрос выполнен успешно».

9.8 Статус Операции по Кассовой ссылке. GET – запрос

Потоки 43, 46.

GET /cash-register-qrc/{qrcId}/{paramsId}

{qrcId} - идентификатор зарегистрированного QR кода в СБП

{paramsId} – идентификатор активного набора параметров

Пример ответа


```
{
  "cashRegisterQrcStatus": "WAITING_PAYMENT",
  "trxStatus": "ACWP",
  "trxId": "A1A2S3D5F6G7H8J9K0C4S5C6D7V5D1K2",
  "amount": "10000",
  "timestamp": "2019-03-29T08:15:07.305Z",
  "payerId": "*********6731",
  "kzo":
"FDOS4JUETLYT639ADAFZ4GAUY9VSM2TG2Y595LQ20EKQF3JM1CIV4ZTZYA1EYIMFMEJSRB2UR7KATMA29QYMY9KLA2NHR
HZ1SKEJ947DTTS67M1GZ6DDUXGJWUXH0Z7X"
}
```


Параметры ответа

Параметр	Описание
cashRegisterQrcStatus	Статус Кассовой ссылки • "INACTIVATED" - Ссылка не активирована • "WAITING_PAYMENT" - Кассовая ссылка СБП активирована и готова к оплате • "IN_PROGRESS" - Операция по Кассовой ссылке СБП в процессе выполнения
trxStatus	Статус операции по Кассовой ссылке • "NTST" - Операция не начата • "RCVD" - Операция в обработке • "ACWP" - Операция завершена успешно • "RJCT" - Операция отклонена
trxId	Идентификатор Операции СБП C2B
amount*	Сумма в копейках
timestamp	Дата и время выполнения Операции
payerId	Маскированный номер телефона Клиента-Плательщика
kzo	Контрольное Значение Операции