

Протокол Litebox автоматической фискализации чеков

[ВЕРСИЯ ДОКУМЕНТА: 0.0.18 ОТ 25.06.2019]

СООТВЕТСТВУЕТ ВЕРСИИ API: V1

1 История изменений

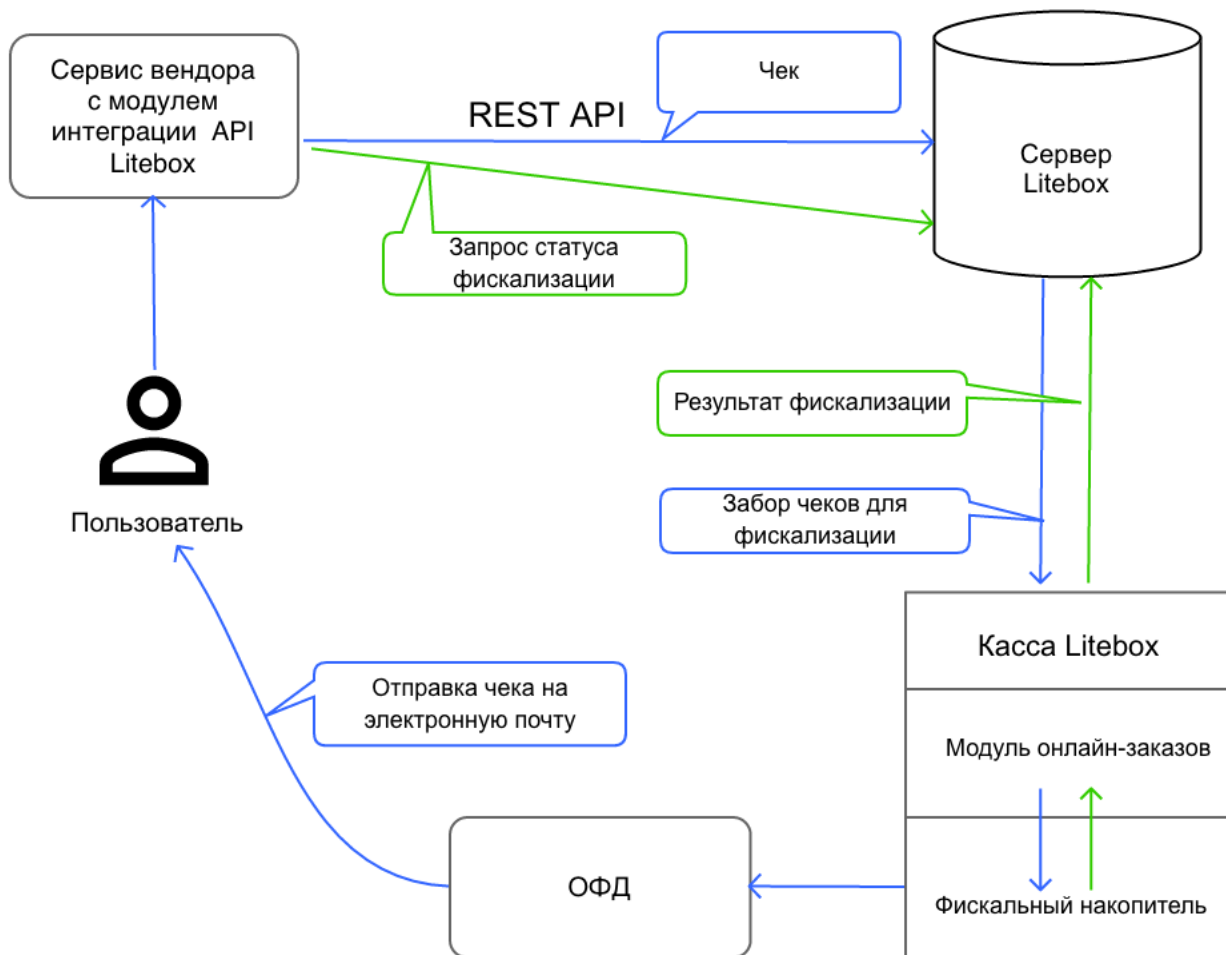
№	Версия документа	Дата	Описание
1.	0.0.1	19.05.2018	- Базовое описание протокола
2.	0.0.2	23.05.2018	- Расширен перечень статус-кодов ошибок - Добавлено описание формата ошибки - В тело чека добавлен параметр payment_address (место расчета/веб-сайт) - Изменено описание статусов чека в очереди фискализации
3.	0.0.3	25.05.2018	- callback_url стал необязательным параметром
4.	0.0.4	11.06.2018	- в методе API /auth параметр login заменен на два параметра: email и phone_number - callback_url исключен из параметров
5.	0.0.5	14.06.2018	- Добавлено Приложение №1 с UI-прототипом формы привязки к сервису Litebox из сервиса Вендора (для метода /auth)
6.	0.0.6	29.09.2018	- Из схемы и формата API удален коллбек клиенту о фискализации (callback_url) - Изменен URL, параметры запроса и ответа эндпойнта фискализации чека - Изменен URL, параметры запроса и ответа эндпойнта статуса операции - Описаны форматы ошибок сервиса и ККТ - Удалены прототипы форм авторизации
7.	0.0.7	04.10.2018	- Исправлены опечатки в примерах запросов - API-метод /auth принимает параметр login вместо email/phone_number - добавлен параметр uuid в объект возвращаемой ошибки error
8.	0.0.8	09.10.2018	- добавлено описание операции «Возврат прихода»
9.	0.0.9	17.10.2018	- удален параметр nomenclature_code (Код товарной номенклатуры) из товарной позиции - удалены налоговые ставки в атрибуте налога на позицию: «vat0» – НДС по ставке 0%; «vat110» – НДС чека по расчетной ставке 10/110; «vat118» – НДС чека по расчетной ставке 18/118. - в запросе статуса операции в объект ошибки error добавлен атрибут uuid
10	0.0.10	26.04.2018	- добавлено описание параметра shop_id - в объект payload статуса фискализации добавлен атрибут receipt_id
11	0.0.11	29.04.2018	- исправлен пример json-тела запроса на фискализацию (параметр “payments”)
12	0.0.12	01.11.2018	- список ошибок ККТ сокращен до 4х
13	0.0.13	23.11.2018	- добавлен http-статус 429 с кодом ошибки 29 «Сервер перегружен запросами»

14	0.0.14	23.01.2019	расширен список значений для атрибутов payment_object, payment_method, vat
15	0.0.15	12.02.2019	расширен список значений для атрибутов payment_object, vat
16	0.0.16	09.02.2019	Документ переработан под общий формат для любого Вендора
17	0.0.17	30.05.2019	Исправлена опечатка именованя поля password в методе авторизации
18	0.0.18	25.06.2019	Добавлен признак печати чека is_print

2 Назначение сервиса.

Цель: обеспечить возможность удаленной регистрации и фискализации чеков (интернет-магазинов, сервисов безналичной оплаты и др).

2.1 Схема процесса фискализации



2.2 Описание процесса фискализации

1. Пользователь взаимодействует с сервисом Вендора услуг, результатом взаимодействия является оформленный чек, требующий фискализации. (Пример: оформление покупки в Интернет-магазине и ее безналичная оплата).
2. Отправка чека сервисом Вендора в заданном формате по REST-API на сервер Litebox. Постановка чека в очередь на обработку кассой. Чеку присваивается уникальный идентификатор (UUID) для последующего опроса состояния чека на сервере.
3. Опрос сервера Litebox кассой на наличие чеков с последующим проведением продажи и фискализации.
4. Касса регистрирует продажу в ФН. ФН отправляет данные в ОФД, который отправляет покупателю чек на электронную почту.
5. Касса отправляет на сервер Litebox уведомление о результате фискализации.
6. Сервер Вендора периодически опрашивает состояние по UUID о статусе фискализации.

3 Описание API Litebox

Для взаимодействия используется REST-API по протоколу HTTPS.

3.1 Базовый URL

```
https://XX.litebox.ru/fiscalization/v1
```

где:

- XX – префикс доменного имени одного из серверов
- («in» – релиз-сервер, «sa-test» - тестовый сервер)
- v1 – текущая версия API

3.2 HTTP-заголовки

Во всех запросах должны присутствовать заголовки:

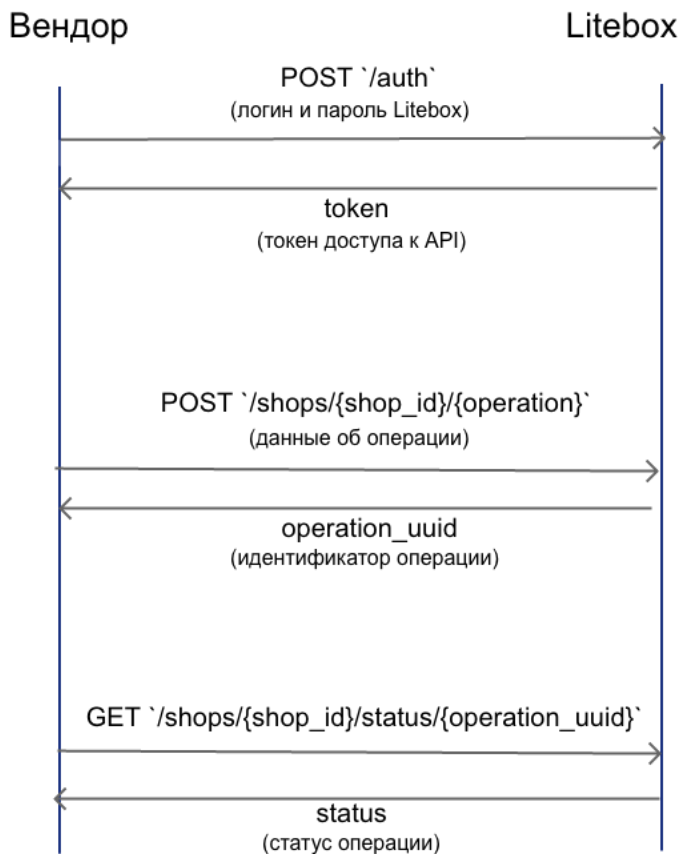
```
Content-type: application/json; charset=utf-8
```

3.3 Версионность API

Сервис поддерживает версионирование. У каждой версии существует идентификатор (v1, v2 и т. д.). При выходе новой версии API предыдущая версия продолжает работать для поддержки обратной совместимости.

При формировании запроса к API всегда необходимо указывать версию, с которой предполагается работать.

3.4 Общая схема работы с REST-API Litebox



3.5 Ошибки сервиса и ККТ

3.5.1 Ошибки сервиса

Код ошибки	Код HTTP	Тип ошибки	Описание
1	400	system	Сервер не смог обработать операцию
10	401	system	Неверный токен аутентификации
12	401	system	Неверный логин или пароль
20	401	system	Код магазина, указанный в URL, для заданного пользователя не найден
29	429	system	Сервер {hostname} перегружен запросами. Повторите попытку позже.
30	400	system	<uuid> операции некорректен или не найден
32	400	system	Ошибка валидации (схемы, структуры, данных запроса)
33	400	system	Документ с заданным <external_id> уже существует в базе магазина <shop_id>
50	500	system	Ошибка сервера
52	502	system	Сервис временно недоступен

3.5.2 Ошибки ККТ

Код HTTP для данных видов ошибок всегда 200, т.к. данный вид ошибок может быть возвращен только при запросе статуса документа.

Код ошибки	Тип ошибки	Описание
1	agent	Нефатальная ошибка Кассового ПО
2	agent	Фатальная ошибка Кассового ПО
-1	driver	Нефатальная ошибка ККТ
-2	driver	Фатальная ошибка ККТ

3.5.3 Формат ошибки

Пример:

```
"error": {  
  "uuid": "4389c882-fd44-4f4b-bce6-89b3f49ebdaa",  
  "code": 1,  
  "type": "system",  
  "text": "Неверный логин или пароль."  
}
```

Параметры:

Название поля	Тип поля	Обязательное	Описание
error	object	Да (в случае ошибки)	Ошибка
└ uuid	string	да	UUID ошибки
└ code	integer	да	Код ошибки
└ type	enum (string)	да	Тип источника ошибки. Возможные значения: - «system» – системная ошибка сервиса - «driver» – ошибка ККТ - «agent» - ошибка Кассового ПО
└ text	string	да	Текст ошибки

4 Описание методов API

Описанные методы подразумевают вызов в контексте базового URL пункта 3.1.

4.1 Авторизация пользователя

Вызов метода авторизации с использованием данных учетной записи Litebox используется для создания связки аккаунта Litebox и Вендора. После вызова метода, в ответе выдается Токен, который однозначно идентифицирует пользователя. Данный токен нужно использовать для всех последующих обращений к API Litebox.

Повторный вызов метода нужно делать при смене токена на стороне Litebox, истечении срока токена, утери или компрометации учетных данных.

POST /auth

Тело запроса в формате JSON:

```
{
  "login": "mylogin@example.com",
  "password": "1M2y3Password"
}
```

Параметры запроса:

Название поля	Тип поля	Обязательное	Описание
login	string	да	Электронная почта/ Номер телефона (как логин учетной записи Litebox/учетной записи МТС соответственно)
password	string	да	Пароль от учетной записи

Параметры ответа:

Название поля	Тип поля	Обязательное	Описание
token	string	да	Уникальный токен, идентифицирующий пользователя

Пример ответа:

```
{
  "token": "9944b09199c62bcf9418ad846dd0e4bbdfc6ee4b"
}
```

4.2 Аутентификация запросов API

Для последующей аутентификации пользователя в HTTP-заголовок Authorization каждого API-запроса необходимо добавить выданный Token учетной записи пользователя (Token-based authentication):

Authorization: Token 9944b09199c62bcf9418ad846dd0e4bbdfc6ee4b

4.3 Операция фискализации чека «Приход»

Постановка чека «Приход» в очередь с целью регистрации и фискализации в ККТ. Авторизационный токен должен быть передан в заголовке запроса.

POST /shops/{shop_id}/sell

Параметр	Тип	Обязательное	Описание
shop_id	number	да	Идентификатор магазина, указанный пользователем в форме подключения в сервисе Вендора

Тело запроса в формате JSON, пример:

```

{
  "external_id": "17052917561851307",
  "timestamp": "01.02.17 13:45:00",
  "receipt": {
    "client": {
      "email": "client@example.com"
    },
    "company": {
      "email": "company@example.com",
      "inn": "1234567891",
      "sno": "osn",
      "payment_address": "https://shopping.ru"
    },
    "items": [
      {
        "name": "Товар весовой",
        "price": 1000,
        "quantity": 0.3,
        "sum": 300.0,
        "measurement_unit": "кг",
        "payment_method": "full_prepayment",
        "payment_object": "commodity",
        "vat": {
          "type": "vat20",
          "sum": 60.0
        }
      },
      {
        "name": "Товар штучный",
        "price": 100,
        "quantity": 1,
        "sum": 100.0,
        "measurement_unit": "шт",
        "payment_method": "full_prepayment",
        "payment_object": "commodity",
        "vat": {
          "type": "vat10",
          "sum": 10.0
        }
      }
    ],
    "payments": [
      {
        "type": "1",
        "sum": 400.0
      }
    ],
    "total": 400.0,
    "is_print": false
  }
}

```

Параметры запроса:

Название поля	Тип поля	Обязательное	Описание
external_id	string	да	Идентификатор чека, уникальный.
timestamp	string	да	Дата и время создания чека в формате: «dd.mm.yyyy HH:MM:SS»
receipt	object	да	Чек
└ client	object	да	Атрибуты клиента
└ email	string	да	Электронная почта покупателя.
└ company	object	да	Атрибуты компании
└ email	string	да	Электронная почта отправителя чека.

└ inn	string	да	ИНН организации. Используется для предотвращения ошибочных регистраций чеков на ККТ
└ sno	enum (string)	да	Система налогообложения. Перечисление со значениями: <ul style="list-style-type: none"> - «osn» – общая СН; - «usn_income» – упрощенная СН (доходы); - «usn_income_outcome» – упрощенная СН (доходы минус расходы); - «envd» – единый налог на вмененный доход; - «esn» – единый сельскохозяйственный налог; - «patent» – патентная СН.
└ payment_address	string	да	Место расчета/URL веб-сайта, на котором произведена покупка
└ items	array of objects	да	Атрибуты позиций
└ name	string	да	Наименование товара.
└ price	number	да	Цена в рублях
└ quantity	number	да	Количество/вес
└ sum	number	да	Сумма в рублях
└ measurement_unit	string	нет	Единица измерения товара, услуги, иного предмета расчета.
└ payment_method	enum (string)	нет Если признак не передан, по умолчанию используется значение «full_prepayment».	Признак способа расчёта. Возможные значения: <ul style="list-style-type: none"> - «full_prepayment» – предоплата 100%. - «credit» – передача в кредит. - «partial_payment» – частичный расчет и кредит. - «full_payment» – полный расчет. - «credit_payment» – оплата кредита. - «prepayment» – предоплата. - «advance» – аванс.
└ payment_object	enum (string)	Нет. Если признак не передан, по умолчанию используется значение «commodity».	Признак предмета расчёта. Возможные значения: <ul style="list-style-type: none"> - «commodity» – товар, за исключением подакцизного товара - «service» – услуга - «payment» – платеж. - «job» – работа. - «excise» – подакцизный товар. - «agent_commission» – агентское вознаграждение. - «gambling_bet» – ставка азартной игры. - «gambling_prize» – выигрыш азартной игры. - «lottery» – лотерейный билет. - «lottery_prize» – выигрыш лотереи. - «intellectual_activity» – предоставление результатов интеллектуальной деятельности. - «composite» – составной предмет расчета. - «another» – иной предмет расчета - «property_right» – имущественное право - «insurance_premium» – страховые взносы - «sales_tax» – торговый сбор - «resort_fee» – курортный сбор. - «non-operating_gain» – внереализационный доход.
└ vat	object	да	Атрибуты налога на позицию
└ type	enum (string)	да	Устанавливает номер налога в ККТ. Перечисление со значениями: <ul style="list-style-type: none"> - «none» – без НДС; - «vat0» – НДС по ставке 0%; - «vat10» – НДС чека по ставке 10%; - «vat110» – НДС чека по расчетной ставке 10/110; - «vat20» – НДС чека по ставке 20%;

			- «vat120» – НДС чека по расчетной ставке 20/120.
└ sum	number	да	Сумма налога позиции в рублях
└ payments	Array of objects	да	Оплаты.
└ type	enum (number)	да	Вид оплаты. Возможные значения: - «1» - электронный
└ sum	number	да	Сумма к оплате в рублях.
└ total	number	да	Итоговая сумма чека в рублях. При регистрации в ККТ происходит расчёт фактической суммы: суммированием значений sum позиций.
└ is_print	boolean	нет Если признак не передан, по умолчанию используется значение «false».	Признак печати чека на чековой ленте Возможные значения: - true: будет распечатан - false: не будет распечатан

Параметры ответа:

Название поля	Тип поля	Обязательное	Описание
uuid	string	да	UUID операции, уникальный
status	enum (string)	да	Имеет следующие значения: - «wait» - документ принят и ожидает в очереди на обработку - «fail» – ошибка
error	object	нет	Ошибка (если отсутствует – значение “null”)
└ code	integer	да	Код ошибки (см. «Ошибки сервиса»)
└ type	enum (string)	да	Тип источника ошибки. Возможные значения: - «system» – системная ошибка сервиса
└ text	string	да	Текст ошибки

Пример ответа:

```
{
  "uuid": "62489ad0-5b98-44f3-ab05-18e161d7db00",
  "status": "wait",
  "error": null
}
```

4.4 Операция фискализации чека «Возврат прихода»

Постановка чека «Возврат прихода» в очередь с целью регистрации и фискализации в ККТ. Авторизационный token должен быть передан в заголовке запроса.

POST /shops/{shop_id}/sell_refund

Параметр	Тип	Обязательное	Описание
shop_id	number	да	Идентификатор магазина, указанный пользователем в форме подключения в сервисе Вендора

Тело, параметры запроса и формат ответа аналогичны операции чека «Приход» (см. выше)

4.5 Запрос статуса операции

Авторизационный token должен быть передан в заголовке запроса.

GET /shops/{shop_id}/status/{operation_uuid}

Параметр	Тип	Обязательное	Описание
shop_id	number	да	Идентификатор магазина, указанный пользователем в форме подключения в сервисе Вендора
operation_uuid	string	да	UUID операции в очереди Litebox, уникальный

Параметры ответа:

Название поля	Тип поля	Обязательное	Описание
uuid	string	да	UUID операции, уникальный
status	enum (string)	да	Имеет следующие значения: - «wait» - документ ожидает в очереди на обработку - «done» – документ успешно обработан ККТ - «fail» – ошибка
device_code	string	да	Код ККТ
error	object	нет	Ошибка (если отсутствует – значение “null”)
└ uuid			UUID ошибки, уникальный
└ code	integer	да	Код ошибки (см. «Ошибки сервиса / Ошибки ККТ»)
└ type	enum (string)	да	Тип источника ошибки. Возможные значения: - «system» – системная ошибка сервиса - «driver» – ошибка ККТ - «agent» - ошибка Кассового ПО
└ text	string	да	Текст ошибки
payload	object	да	Реквизиты фискализации документа
└ fiscal_receipt_number	integer	да	Номер чека в смене
└ shift_number	integer	да	Номер смены
└ receipt_datetime	string	да	Дата и время документа из ФН
└ total	number	да	Итоговая сумма документа в рублях
└ fn_number	string	да	Номер ФН
└ ecr_registration_number	string	да	Регистрационный номер ККТ
└ fiscal_document_number	integer	да	Фискальный номер документа
└ fiscal_document_attribute	integer	да	Фискальный признак документа
└ fns_site	string	да	Адрес сайта ФНС
└ receipt_id	string	нет	Внутренний идентификатор чека в системе Litebox

Поля **payload** и **device_code** равны **null** в случае наличия ошибки **error**.

Пример ответа (200 OK):

```
{
  "uuid":"00423d40-9ab7-4395-bead-59819f770503",
  "error":null,
  "status":"done",
  "payload":{
    "total":400,
    "fns_site":"www.nalog.ru",
    "fn_number":"2345555565656",
    "shift_number":14,
    "receipt_datetime":"01.01.2018 21:00:00",
    "fiscal_receipt_number":4,
    "fiscal_document_number":34,
    "ecr_registration_number":"3453453535345",
    "fiscal_document_attribute":776866786868
  },
  "device_code":"KKT LB.01-2-34"
}
```

Пример ответа с ошибкой (**200 OK**):

```
{
  "uuid":"00423d40-9ab7-4395-bead-59819f770503",
```

```
"error": {  
  "uuid": "62489ad0-5b98-44f3-ab05-18e161d7db00",  
  "code": 2,  
  "type": "agent",  
  "text": "Документ не может быть обработан данной ККТ, так как она зарегистрирована с другим ИНН или адресом  
расчёта",  
},  
"status": "fail",  
"payload": null,  
"device_code": null  
}
```