

# **Описание API системы телеметрии «Эфор»**

Версия 2.0

## Оглавление

1	Авторские права.....	4
2	Общие положения.....	4
2.1	Команда получения данных (GET).....	4
2.2	Формат URL команды получения данных.....	5
2.3	Кодирование значений параметров.....	6
2.4	Фильтры команд получения данных.....	6
2.4.1	Фильтр по значению параметра.....	7
2.4.2	Фильтр по диапазону.....	7
2.5	Формат ответа на команду получения данных.....	7
2.6	Постраничная выборка данных.....	7
2.7	Команда изменения данных (POST).....	8
3	Основные операции с данными.....	10
3.1	Команда.....	10
3.2	Таблица.....	10
3.3	Особенность команд POST с таблицами.....	11
3.4	Отчет.....	11
4	Форматы данных.....	12
5	Аутентификация.....	13
5.1	Команда Login.....	13
5.2	Команда Logout.....	13
6	Модели торговых автоматов.....	14
6.1	Команда запроса моделей торговых автоматов.....	14
7	Кассы.....	15
7.1	Таблица касс.....	15
7.2	Чеки.....	17
8	Шаблоны торговых автоматов.....	18
8.1	Таблица шаблонов.....	18
8.2	Таблица продуктов шаблона.....	19
9	Конфигурации торговых автоматов.....	20
9.1	Номенклатура.....	20
9.2	Таблица конфигураций.....	20
9.3	Команда запроса конфигураций.....	23
9.4	Таблица прайс-листов.....	24
9.5	Таблица продуктов.....	25
10	Торговые автоматы.....	27
10.1	Команда запроса торговых автоматов.....	27
10.2	Команда редактирования ТА.....	28
10.3	Команда привязки конфигурации к ТА.....	29
10.4	Команда отвязывания конфигурации от ТА.....	29
10.5	Команда добавления точки для ТА.....	29
10.6	Команда чтения точек для ТА.....	30
10.7	Команда отвязывания точки от ТА.....	30
10.8	Команда изменения точки.....	31

11	Устройства автомата.....	32
11.1	Команда запроса одного устройства.....	32
11.2	Команда запроса информации о устройстве.....	33
12	Аудиты торговых автоматов.....	33
12.1	Учет наличных.....	33
12.2	Учет продаж по факту продажи.....	34
12.3	Таблица аудитов.....	34
12.4	Счетчики продаж по продуктам.....	36
12.5	Аудиты со счетчиками продаж по продуктам.....	38
12.6	Аудит в формате EVADTS.....	41
11.7	Аудит статистики.....	41
13	Загрузки торговых автоматов.....	42
13.1	Загрузки.....	42
14	Команды модему.....	44
14.1	Команды модему.....	44
15	Проведение платежа через внешнее приложение.....	44
15.1	Создание транзакции продажи.....	44
15.2	Получения состояния транзакции.....	45
16	Отчеты.....	46
16.1	Отчет по товарам.....	46
16.2	Отчет продаж от аудита до заданной даты.....	47
16.3	Отчет по событиям заданного автомата.....	49
16.4	Отчет по списку чеков.....	49
16.5	Получение списка модемов.....	51
16.6	Отчёт «Состояние торговых автоматов».....	52

## 1 Авторские права

Данное API является объектом авторских прав компании «Эфор».

Данное API может свободно использоваться для интеграции с системой телеметрии «Эфор», в частности для выгрузки данных и внесения изменений.

Данное API или принципы, изложенные в данном документе, не могут быть использованы для реализации в других системах телеметрии без письменного согласия компании «Эфор».

Текст документа не может быть использован при написании документов других компаний или физических лиц без письменного согласия компании «Эфор».

## 2 Общие положения

Адрес сервера <https://erp.ephor.online>

Протокол передачи данных HTTPS.

Пример работы с соединением в 1с вы можете найти по ссылке.

<https://its.1c.ru/db/metod8dev#content:5574:hdoc>

Любые команды выполняются от имени пользователя. В системе предусмотрено разграничение прав доступа. Поэтому любой обмен данными с сервером состоит из следующих команд:

1. login (создается сессия пользователя)
2. команда1
3. команда2
4. logout (сессия завершается)

Все команды делятся на команды изменения данных и получения данных.

То что написано далее предполагает, что вы знаете структуру HTTP запроса. Если вы не знаете устройство HTTP запроса, то настоятельно рекомендуется изучить общие положения.

### 2.1 Команда получения данных (GET)

Для запроса данных с сервера необходимо серверу сообщить следующие данные:

- путь к обработчику команды;
- тип команды (получение или изменение данных);
- команду;
- идентификатор сессии, в рамках которой будет выполняться команда;
- фильтры записей;
- способ сортировки записей;
- параметры страничной выборки.

Рассмотрим как это делать на примере команды получения списка конфигураций. Ниже

приведены ключевые заголовки HTTP запроса:

```
GET /api/2.0/config/Config.php?action=ReadConfig&_dc=1518591517187
&start=0&limit=25
&sort=%5B%7B%22property%22%3A%22name%22%2C%22direction%22%3A%22ASC%22%7D%5D
&filter=%5B%7B%22property%22%3A%22automat_model_id%22%2C%22value%22%3A
%221%22%7D%2C%7B%22property%22%3A%22state%22%2C%22value%22%3A1%7D%5D
HTTP/1.1

Host: erp.ephor.online
Connection: keep-alive
Cookie: PHPSESSID=oa2tciltpg16scaqe1fkrjoq63Request
```

В первой строке заданы:

- Метод запроса. В данном случае «GET». GET используется для получения данных, а POST используется для изменения данных.
- URL-запроса. Смотрите ниже детальное описание в главе «Формат URL команды».
- Версия протокола. В данном случае это «HTTP/1.1».

Во второй строке указан домен сервера. Его необходимо указывать для того, чтобы HTTP-сервер понимал к какому сайту вы обращаетесь.

В третьей строке серверу сообщается, что соединение рвать не надо и далее по этому соединению будут посланы ещё команды. Этот режим работы рекомендуется, но не обязателен.

В четвертой строке серверу передается идентификатор сессии. Его мы получаем в ответе на команду «Login». То есть после того как сервер, проверив логин и пароль, нас идентифицирует и убедится, что мы это мы, сервер создаст сессию с нашими правами доступа и присвоит ей уникальный идентификатор «PHPSESSID». Каждый раз когда сервер будет получать запрос с данным идентификатором он будет понимать кто запросил выполнение команды и какими правами он обладает. Если около получаса не посылать команды, то сессия будет закрыта и потребуются повторное выполнение команды «Login».

## 2.2 Формат URL команды получения данных

Рассмотрим детально, что передается в URL HTTP-запроса:

```
/api/2.0/config/Config.php?action=ReadConfig&_dc=1518591517187
&start=0&limit=25
&sort=%5B%7B%22property%22%3A%22name%22%2C%22direction%22%3A%22ASC%22%7D%5D
&filter=%5B%7B%22property%22%3A%22automat_model_id%22%2C%22value%22%3A
%221%22%7D%2C%7B%22property%22%3A%22state%22%2C%22value%22%3A1%7D%5D
```

Данный URL состоит из следующих частей:

- путь: /api/2.0/config/Config.php Путь к обработчику запроса.
- параметр «action=ReadConfig» Команда, которую сервер должен выполнить.
- параметр «\_dc=1518591517187» Уникальный идентификатор запроса. Лучше всего использовать текущее время в секундах. Необходим для защиты от повторяющихся запросов.
- параметры «start», «limit» описаны отдельно. Отвечают за постраничную выборку

данных. Смотрите ниже главу «Постраничная выборка данных».

- параметр «sort» отвечает за сортировку записей в ответе. Смотрите ниже главу «Кодирование значений параметров».
- Параметр «filter» отвечает за то, какие записи будут выбраны. Смотрите ниже главу «Кодирование значений параметров».

## 2.3 Кодирование значений параметров

Рассмотрим детально, как задано значение фильтра:

```
filter=%5B%7B%22property%22%3A%22automat_model_id%22%2C%22value%22%3A%221%22%7D%2C%7B%22property%22%3A%22state%22%2C%22value%22%3A%221%7D%5D
```

Как мы можем видеть значение параметра «filter» закодировано. Если необходимо передать значение параметра в URL, то название параметра и его значение по требованиям HTTP-протокола не должны содержать никаких символов, кроме букв, цифр и подчеркивания. Если они содержат другие символы, то эти символы заменяются на их шестнадцатеричные коды.

Вот некоторые коды символов:

Символ	Код
[	%5B
]	%5D
{	%7B
}	%7D
"	%22
:	%3A
,	%2C

В данном случае передаваемое значение фильтра имеет вид:

```
[{"property":"automat_model_id","value":"1"}, {"property":"state","value":1}]
```

Здесь закодировано два фильтра: фильтр модели ТА «automat\_model\_id» со значением 1 и фильтр состояния конфигурации «state» со значением 1.

В данном случае передаваемое значение сортировки имеет вид:

```
[{"property":"name","direction":"ASC"}]
```

Здесь закодирована сортировка по возрастанию значения поля «name».

## 2.4 Фильтры команд получения данных

В системе предусмотрено несколько разных фильтров записей:

- Фильтр по значению параметра;
- Фильтр по диапазону.

### 2.4.1 Фильтр по значению параметра

При применении фильтра по значению параметра сервер возвращает только те записи значения параметров, которых совпадают со значением или значениями заданными в этом фильтре.

Например, нам нужно отфильтровать аудиты типа 0.

Для этого мы должны задать следующий фильтр в запрос:

```
{"property":"type","value":0}
```

Если же нам нужно отфильтровать аудиты нескольких типов одновременно, например 0 и 1, то мы должны задать в качестве значения фильтра список значений.

```
{"property":"type","value":[0,1]}
```

### 2.4.2 Фильтр по диапазону

При применении фильтра по диапазону параметра сервер возвращает только те записи значение соответствующего параметра которых находится в пределах заданного диапазона. Значение параметра может быть как внутри диапазона, так и совпадать с его границами.

Например, нам нужно отфильтровать аудиты с даты 01.01.2018 по 20.01.2018, то нужно задать два параметра в список фильтров:

```
{"property":"date_from","value":"2018-01-01 00:00:00"},  
{"property":"date_to","value":"2018-01-20 23:59:59"}
```

Названия параметров границ будут заданы в описании фильтра.

## 2.5 Формат ответа на команду получения данных

Ответ на GET команды приходит в теле сообщения HTTP-ответа в формате JSON и выглядит следующим образом:

```
{"success":true,"total":2,"data": [  
  {"id":4,"automat_model_id":1,"name":"test2","state":1},  
  {"id":18,"automat_model_id":1,"name":"test siba 3","state":1} ]}
```

где

- `success` – результат выполнения команды. `True` – команда выполнена успешно. `False` – в ходе выполнения команды произошла ошибка.
- `total` – общее количество записей, который подходят под запрошенные фильтры. В ответе могут быть не все записи.
- `data` – список записей ответа.

Если при выполнении запроса происходит ошибка, то ответ будет выглядеть аналогично ответу на команду POST.

## 2.6 Постраничная выборка данных

Количество записей, подходящих под заданные фильтры может исчисляться миллионами, а такое количество невозможно передать одним запросом. Поэтому, по умолчанию размер выборки данных с сервера ограничен 1000 записей.

Как это работает? При обработке запроса на получение данных сервер всегда подсчитывает общее количество записей соответствующих запросу, но возвращает не более 1000. Общее количество записей соответствующих запросу возвращается в виде значения параметра ответа «total».

В системе предусмотрена выгрузка данных постранично. Для этого и нужны поля «start» и «limit». Поле «start» задает порядковый номер записи, начиная с которой нужно возвращать очередную порцию данных. Поле «limit» указывает сколько записей нужно вернуть в этот раз.

Крайне рекомендуется также задать и способ сортировки записей, так как одинаковый порядок сортировки записей в разных запросах не гарантируется сервером. Другими словами, если не задать сортировку явно, то одни и те же записи могут попасть в разные страницы или не попасть ни в одно из них.

## 2.7 Команда изменения данных (POST)

Для изменения данных сервера необходимо серверу сообщить следующие данные:

- путь к обработчику команды;
- тип команды (получение или изменение данных);
- команду;
- параметры команды;
- идентификатор сессии, в рамках которой будет выполняться команда;
- уникальный идентификатор команды (для защиты от повторных выполнений).

Рассмотрим параметры запроса на изменение данных на примере команды создания конфигурации.

Ключевые заголовки HTTP запроса:

```
POST /api/2.0/config/Config.php?action=CreateConfig&_dc=1518593389938
HTTP/1.1
Host: erp.ephor.online
Connection: keep-alive
Content-Length: 108
Cookie: PHPSESSID=oa2tciltpg16scaqe1fkpjoq63GET
```

В первой строке указаны:

- Метод запроса — POST.
- URL-путь: /api/2.0/config/Config.php Путь к обработчику запроса.
- URL-параметр: action=CreateConfig Команда, которую сервер должен выполнить.
- URL-параметр: \_dc=1518593389938 Уникальный идентификатор запроса. Лучше всего использовать текущее время в секундах. Необходим для защиты от повторяющихся запросов.

В строке «Content-Length» указан размер данных, которые находятся в теле сообщения после заголовков HTTP-запроса. В теле сообщения передаются данные команды, которые сервер должен использовать при выполнении команды. Как правило это поля записей в

JSON-формате.

В данном случае тело сообщения имеет вид:

```
{"name":"1", "automat_model_id":7, "automat_model_name":"","state":1}
```

Ответ на POST команды приходит в теле сообщения HTTP-ответа в формате JSON и выглядит следующим образом:

```
{"success":true,"data":  
[{"id":"19","automat_model_id":7,"name":"1","state":1}]}
```

Ключевое здесь значение поля «success». Если оно равно true, то команда выполнена без ошибок. Если оно равно false, то команда не была выполнена.

Если команда предполагает создание записи в базе данных, то в ответе будут переданы поля созданной записи. Нас в первую очередь интересует поле «id», так как часто с созданной записью требуется сделать ещё что-нибудь.

## 3 Основные операции с данными

Все операции с данными делятся на следующие группы:

- отдельные команды (далее «команда»);
- операции с таблицами (далее «таблица»);
- отчеты (далее «отчет»).

### 3.1 Команда

Далее для каждой команды мы будем указывать:

- Метод (POST или GET);
- Путь;
- Команда (action);
- Фильтры (только для метода GET);
- Таблицу параметров (для метода POST и GET, со следующими столбцами: «Название» - название параметра, «Тип» - формат значения параметра, «Назначение» - описание параметра).

### 3.2 Таблица

Большинство данных представлено в виде таблиц.

С данными каждой таблицы возможны следующие операции:

- (C)REATE – создание записи (метод POST, параметры);
- (R)EAD – чтение записей (метод GET, фильтры);
- (U)PDATE – обновление записи (метод POST, идентификатор записи, параметры);
- (D)ELETE – удаление записи (метод POST, идентификатор).

Так как операции выполняются с одними и теми же записями, поэтому множества параметров пересекаются. Далее в отдельном столбце мы будем указывать первые буквы команд, с которыми данный параметр необходимо использовать.

Соответственно для каждой таблицы мы будем указывать:

- Путь (все операции с одной таблицей имеют один и тот же путь);
- Команды (полные названия всех операций, который необходимо задавать как URL-параметр «action»);
- Таблицу параметров (со следующими столбцами: «Название» - название параметра, «Тип» - формат значения параметра, «Назначение» - описание параметра, «Применение» - первые буквы команд, с которыми данный параметр необходимо использовать);
- Фильтры (используются только с командой READ).

### 3.3 Особенность команд POST с таблицами

В системе предусмотрена возможность выполнения нескольких однотипных команд за один запрос. Например нам нужно создать несколько прайс-листов конфигурации. Это можно сделать как отдельными запросами, так и за один запрос. Для этого в теле запроса нужно задать несколько наборов параметров:

```
POST /api/2.0/config/Config.php?action=CreateConfigPriceList&_dc=1519057132529
HTTP/1.1
```

```
Host: local.ephor.online
```

```
Connection: keep-alive
```

```
Content-Length: 205
```

```
Content-Type: application/json
```

```
Cookie: PHPSESSID=dvupmu8p4okm3eu24s624o5fpv
```

```
[{"config_id":1,"payment_device":"CA","number":0},
{"config_id":1,"payment_device":"DA","number":1}]
```

### 3.4 Отчет

Далее для каждого отчета мы будем указывать:

- Метод (как правило GET);
- Путь;
- Команда (action);
- Фильтры (только для метода GET);
- Таблицу параметров (со следующими столбцами: «Название» - название параметра, «Тип» - формат значения параметра, «Назначение» - описание параметра).

## 4 Форматы данных

Число — только цифры;

Массив — Пример [0,1];

Строка — строка символов. Печатные символы, коды которых совпадают с кодами ASCII от 0 до 127, передаются как есть. Остальные символы должны передаваться закодированными в формате UTF-16. Формат кодирования символа выглядит следующим образом \uxxxx, где xxxx – шестнадцатичная запись кода UTF-16 символа.

Дата — дата и время. Формат записи «YYYY-MM-DD hh:mm:ss», где YYYY – года, MM – месяц (от 1 до 12), DD – день (от 1 до 31), hh – час (от 0 до 23), mm – минута (от 0 до 59), ss – секунда (от 0 до 59). Пример «2018-01-01 02:02:20». В JSON задается как строка.

Время — только время. Формат записи «hh:mm:ss», где mm – минута (от 0 до 59), ss – секунда (от 0 до 59). Пример «2018-01-01 02:02:20». В JSON задается как строка.

BOOL – бинарное значение: true или false.

## 5 Аутентификация

В системе любые запросы на получение и изменение данных осуществляются только известными системе пользователями. Для опознания системой пользователя необходимо сообщить системе имя пользователя (login). Для подтверждения, что вы являетесь тем пользователем, логин которого вы передали системе, необходимо передать пароль (password).

### 5.1 Команда Login

Метод: POST

Путь: /api/2.0/Auth.php

Команда: Login

Параметры:

Название	Тип	Назначение
login	строка	
password	строка	
time_zone	число	Смещение часового пояса (по умолчанию -3) - Москва

### 5.2 Команда Logout

Метод: POST

Путь: /api/2.0/Auth.php

Команда: Logout

Параметры: нет

## 6 Модели торговых автоматов

### 6.1 Команда запроса моделей торговых автоматов

Метод: GET

Путь: /api/2.0/automat/Automat.php

Команда: ReadAutomatModel

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
type	число	Тип торгового автомата: 0 — кофейный; 1 — снековый; 2 — хорека; 3 — газированная вода.
name	строка	Название модели торгового автомата

Фильтры:

Название	Тип	Значение
type	число	Тип торгового автомата

## 7 Кассы

### 7.1 Таблица касс

Путь: /api/2.0/fr/Fr.php

Команды: CreateFr, ReadFr, UpdateFr, DeleteFr

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
name	строка	Название	CRU
password	строка	пароль	CRU
login	строка	логин	CRU
phone	строка	телефон	CRU
email	строка	почта	CRU
type	число	Тип кассы: 1 — PayOnline-01-ФА 2 — Казначей ФА 4 — Терминал ФА 5 — OrangeData 6 — Chekonline 7 — OrangeData + Эфор 9 — NanoKassa 10 — Server OrangeData 11 — Server OrangeData + Эфор 12 — OFD	CRU
dev_interface	число	Интерфейса подключения кассы: 0 — RS-232 1 — ETHERNET (для локальных) 2 — GSM (для облачных)	CRU
dev_addr	строка	Адрес кассы. Для локальных касс должен быть всегда 192.168.1.210. Для облачных касс здесь задается доменное имя сервиса.	CRU
dev_port	число	Порт кассы или сервиса.	CRU
ofd_addr	строка	Адрес ОФД. Задается только для локальных касс. Задаваться должен только IP-адрес, а не доменное имя!	CRU
ofd_port	число	Порт ОФД. Задается только для локальных касс.	CRU
inn	строка	ИНН организации. Задается только для OrangeData и ServerOrangeData.	CRU
param1	строка	Доп параметр. Только для OrangeData и	CRU

		EphorOrangeData и ServerOrangeData – это группа касс.	
auth_public_key	строка	Публичный ключ для аутентификации в PEM-формате. Задается только для OrangeData и ServerOrangeData.	CRU
auth_private_key		Закрытый ключ для аутентификации в PEM-формате без пароля. Задается только для OrangeData и ServerOrangeData.	CRU
sign_private_key		Закрытый ключ для подписи в PEM-формате без пароля. Задается только для OrangeData и ServerOrangeData.	CRU

Фильтры: нет

Примечание:

- Привязка касс осуществляется при создании или редактировании торгового автомата.

## 7.2 Чеки

Метод: POST

Путь: /api/2.0/report/sale/Sale.php

Команда: Fiscal

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи продажи	RUD
automat_id	число	Идентификатор автомата	CRU
date	строка	Дата и время продажи номенклатуры	CRU
name	строка	Наименование номенклатуры	CRU
dataCheck	строка	Данные по ответу от кассы	CRU

Фильтры:

Название	Тип	Значение
id	Массив чисел	Идентификатор записи продажи
from	строка	Дата и время начала периода
to	строка	Дата и время окончания периода

## 8 Шаблоны торговых автоматов

Шаблон торговых автоматов задают количество и названия продуктов, а также привязку продуктов к номерам кнопок.

### 8.1 Таблица шаблонов

Путь: /api/2.0/template/Template.php

Команды: CreateTemplate , ReadTemplate , UpdateTemplate , DeleteTemplate

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
name	строка	Название	CRU
automat_model_id	число	Идентификатор модели ТА	CRU
payment_type	число	Платежная шина: EXECUTIVE – 1 MDB – 2	CRU
currency	число	Валюта: 1643 — Рубли (рекомендуется) 1810 — Рубли до 1998	CRU
decimal_point	число	Положение десятичной точки: 0, 1 или 2.	CRU
max_credit	число	Максимальный кредит в рублях.	CRU
tax_system	число	Система налогообложения: 0x01 — Общая ОСН; 0x02 — Упрощенная доход; 0x04 — Упрощенная доход минус расход; 0x08 — Единый налог на вмененный доход; 0x10 — Единый сельскохозяйственный налог; 0x20 — Патентная система налогообложения.	CRU
comment	строка	Комментарий	CRU
state	строка	Активность записи: 0 — незавершенная 1 — активная 2 — устаревшая	CRU
price_holding	число	Режим MDB PriceHolding: 0 – отключено 1 — включено	CRU
multivend	число	Режим мультипродажи: 0 – отключено 1 — включено	CRU
credit_holding	число	Режим удержания кредита:	CRU

		0 – отключено 1 — включено	
--	--	-------------------------------	--

Фильтры:

Название	Тип	Значение
id	число	Смотрите одноименный параметр
automat_model_id	число	Смотрите одноименный параметр
payment_type	число	Смотрите одноименный параметр
state	число	Смотрите одноименный параметр

## 8.2 Таблица продуктов шаблона

Путь: /api/2.0/template/Template.php

Команды: CreateTemplateProduct, ReadTemplateProduct, UpdateTemplateProduct, DeleteTemplateProduct

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
template_id	число	Идентификатор шаблона	CRU
select_id	число	Идентификатор продукта. Это значение используется в EVADTS отчете и в конфигурации для идентификации продукта.	CRU
cashless_id	число	Номер, возвращаемый автоматом при выборе данного продукта.	CRU
tax_rate	число	Налоговая ставка: 0 — без НДС; 1 — НДС 0%; 2 — НДС 10%; 3 — НДС 20%. 4 — НДС 12%. 5 — НДС 5%. 6 — НДС 7%. 7 — НДС 5/105. 8 — НДС 7/107. 9 — НДС 10/110. 10 — НДС 20/120. 11 — НДС 22%. 12 — НДС 22/122.	CRU

Фильтры:

Название	Тип	Значение
id	число	Смотрите одноименный параметр

## 9 Конфигурации торговых автоматов

Как правило в любой сети торговых автоматов есть несколько типовых шаблонов настроек. В данной системе такой шаблон называется «Конфигурацией». Каждая конфигурация включает список прайс-листов, список продуктов и матрицу цен (для каждого продукта задается цена в рамках каждого прайс-листа).

Ниже будут описаны соответствующие таблицы:

### 9.1 Номенклатура

Путь: /api/2.0/ware/Ware.php

Команды: CreateWare, ReadWare, UpdateWare, DeleteWare

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
code	строка	Артикул. Значение должно быть уникальным. Если задано значение NULL, то артикул считается незадаанным.	CRU
name	строка	Название товара.	CRU
type	число	Тип товара 0 — не задан 1 — рецепт 2 — снэк	CRU
state	число	Актуальность товара 0 — в разработке 1 — активный 2 — устаревший Только «активные» товары будут выводиться вы выпадающих списках при настройке ассортимента	CRU

Фильтры:

Название	Тип	Значение
id	число	Смотрите одноименный параметр
code	строка	Смотрите одноименный параметр
type	число	Смотрите одноименный параметр
state	число	Смотрите одноименный параметр

### 9.2 Таблица конфигураций

Путь: /api/2.0/config/Config.php

Команды: CreateConfig, ReadConfig, UpdateConfig, DeleteConfig

## Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
name	строка	Название конфигурации	CRU
automat_model_id	число	Идентификатор модели торгового автомата, с которой необходимо связать запись	CR
payment_type	число	Способ подключения телеметрии: 1 — MDB-MASTER; 2 — EXE-SLAVE; 3 — EXE-MASTER; 4 — MDB-SLAVE.	CR
dex_port	число	Выгрузка EVADTS из внешних источников: 0 — выключена (по умолчанию); 1 — включена.	CR
template_id	число	Идентификатор шаблона, на основе которого была создана конфигурации. Если задан null, то будет выбран системный шаблон.	CR
currency	число	Валюта: 1643 — Рубли (рекомендуется) 1810 — Рубли до 1998	CRU
decimal_point	число	Положение десятичной точки: 0, 1 или 2.	CRU
max_credit	число	Максимальный кредит в рублях.	CRU
show_change	число	Работает только в EXE-MASTER: 1 — показывать нехватку размена; 0 — игнорировать нехватку размена.	CRU
cl_num	число	Номер картридера: 1 или 2(по умолчанию)	CRU
cl_sf	число	Множитель: число от 0 до 255 , 1 (по умолчанию)	CRU
category_money	число	Подробные события : 0 — выключены (по умолчанию); 1 — включены.	CRU
tax_system	число	Система налогообложения: 0x01 — Общая ОСН; 0x02 — Упрощенная доход; 0x04 — Упрощенная доход минус расход; 0x08 — Единый налог на вмененный доход; 0x10 — Единый сельскохозяйственный налог; 0x20 — Патентная система налогообложения.	CR
state	число	Состояние конфигурации 0 — незавершенная	CRU

		1 — активная 2 — устаревшая	
price_holding	число	Режим MDB PriceHolding: 0 — отключено 1 — включено	CRU
multivend	число	Режим мультипродажи: 0 — отключено 1 — включено	CRU
credit_holding	число	Режим удержания кредита: 0 — отключено 1 — включено	CRU
products	список	Необязательный параметр команды CreateConfig. Список параметров продуктов. Если задан, то будет наложен на выбранный шаблон. Подробности смотрите ниже.	

Фильтры:

Название	Тип	Значение
id	число	Смотрите одноименный параметр
automat_model_id	число	Смотрите одноименный параметр
state	число	Смотрите одноименный параметр

Примечание:

- Если при создании конфигурации задан шаблон, то список продуктов создается автоматически. Пользователю остается только задать номенклатуру и цены.

Параметры продуктов при создании конфигурации:

Название	Тип	Назначение	Применение
select_id	строка	Идентификатор кнопки. Эти идентификаторы должны совпадать с номерами в EVADTS отчете.	CRU
ware_id	число	Идентификатор товара из списка номенклатуры. Если задан NULL, то в качестве названия товара используется значение поля «name». Если задан и поле «name» пустое, то в качестве названия товара используется название товара из списка номенклатуры.	CRU
cid	строка	Код кнопки	CRU
type	число	Тип товара 0 — не задан 1 — рецепт 2 — снэк	CRU
qt_max	число	Максимальное количество товара в ячейке	CRU

qt_recom	число	Рекомендуемое количество товара в ячейке	CRU
name	строка	Название продукта. Если задана пустая строка и задан «ware_id», то в качестве названия товара используется название товара из списка номенклатуры, иначе используется данное название.	CRU
tax_rate	число	Налоговая ставка: 0 — без НДС; 1 — НДС 0%; 2 — НДС 10%; 3 — НДС 20%. 4 — НДС 12%. 5 — НДС 5%. 6 — НДС 7%. 7 — НДС 5/105. 8 — НДС 7/107. 9 — НДС 10/110. 10 — НДС 20/120. 11 — НДС 22%. 12 — НДС 22/122.	CRU
pl_<pl_id>	число	Цена данного продукта в рамках прайс-листа, название которого необходимо вставить вместо «pl_id». Например, у нас есть прайс-листы с названиями CA0 и DA1, то необходимо для каждого задать цены через параметры с названиями pl_CA0 и pl_DA1	CRU

### 9.3 Команда запроса конфигураций

Метод: GET

Путь: /api/2.0/config/Config.php

Команда: ReadConfigDetail

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
name	строка	Название конфигурации
automat_model_id	число	Идентификатор модели торгового автомата
automat_model_name	строка	Название модели торгового автомата
payment_type	число	Способ подключения телеметрии: 1 — MDB-MASTER; 2 — EXE-SLAVE; 3 — EXE-MASTER; 4 — MDB-SLAVE.

dex_port	число	Выгрузка EVADTS из внешних источников: 0 — выключена; 1 — включена.
template_id	число	Идентификатор шаблона, на основе которого была создана конфигурации. Если при создании конфигурации не использовался шаблон, то должен быть равен null.
currency	число	Валюта: 1643 — Рубли (рекомендуется) 1810 — Рубли до 1998
decimal_point	число	Положение десятичной точки: 0, 1 или 2.
max_credit	число	Максимальный кредит в рублях.
show_change	число	Работает только в EXE-MASTER: 1 — показывать нехватку размена; 0 — игнорировать нехватку размена.
tax_system	число	Система налогообложения: 0x01 — Общая ОСН; 0x02 — Упрощенная доход; 0x04 — Упрощенная доход минус расход; 0x08 — Единый налог на вмененный доход; 0x10 — Единый сельскохозяйственный налог; 0x20 — Патентная система налогообложения.
state	число	Состояние конфигурации
used	BOOL	True – конфигурация используется, false – конфигурация не используется. Если конфигурация привязана хотя бы к одному торговому автомату, то её редактирование запрещено на уровне базы данных.

Фильтры:

Название	Тип	Значение
id	число	Смотрите одноименный параметр
automat_model_id	число	Смотрите одноименный параметр
state	число	Смотрите одноименный параметр

## 9.4 Таблица прайс-листов

Путь: /api/2.0/config/Config.php

Команды: CreateConfigPriceList, ReadConfigPriceList, UpdateConfigPriceList, DeleteConfigPriceList

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
config_id	число	Идентификатор конфигурации, с которой	CR

		необходимо связать прайс-лист	
payment_device	строка	Устройство оплаты (латинские буквы) CA – наличные DA – картридер1 DB – картридер2	CRU
number	число	Номер прайс-листа	CRU
work_week	число	Дни недели, когда прайс-лист работает. Дни задаются установкой битов числа: бит 0 — воскресенье бит 1 — понедельник бит 2 — вторник бит 3 — среда бит 4 — четверг бит 5 — пятница бит 6 — суббота	CRU
work_time	время	Время, с которого начинает работать прайс-лист	CRU
work_interval	число	Длительность действия прайс-листа в секундах после достижения заданного времени.	CRU

Фильтры:

Название	Тип	Значение
config_id	число	Идентификатор конфигурации

Примечание:

- прайс-листы CA0 и DA1 создаются автоматически при создании конфигурации;
- параметры work\_week, work\_time и work\_interval работают только в режиме подключения MDB-EXE.

## 9.5 Таблица продуктов

Путь: /api/2.0/config/Config.php

Команды: CreateConfigProduct, ReadConfigProduct, UpdateConfigProduct, DeleteConfigProduct

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RUD
config_id	число	Идентификатор конфигурации, с которой необходимо связать прайс-лист	CR
select_id	строка	Идентификатор кнопки. Эти идентификаторы должны совпадать с номерами в EVADTS отчете.	CRU
cid	число	Номер, возвращаемый автоматом при выборе	CRU

		данного продукта.	
ware_id	число	Идентификатор товара из списка номенклатуры. Если задан NULL, то в качестве названия товара используется значение поля «name». Если задан и поле «name» пустое, то в качестве названия товара используется название товара из списка номенклатуры.	CRU
type	число	Тип товара 0 — не задан 1 — рецепт 2 — снэк	CRU
qt_max	число	Максимальное количество товара в ячейке	CRU
qt_recom	число	Рекомендуемое количество товара в ячейке	CRU
name	строка	Название продукта. Если задана пустая строка и задан «ware_id», то в качестве названия товара используется название товара из списка номенклатуры, иначе используется данное название.	CRU
tax_rate	число	Налоговая ставка: 0 — без НДС; 1 — НДС 0%; 2 — НДС 10%; 3 — НДС 20%. 4 — НДС 12%. 5 — НДС 5%. 6 — НДС 7%. 7 — НДС 5/105. 8 — НДС 7/107. 9 — НДС 10/110. 10 — НДС 20/120. 11 — НДС 22%. 12 — НДС 22/122.	CRU
pl_<pl_id>	число	Цена данного продукта в рамках прайс-листа, идентификатор которого необходимо вставить вместо «pl_id». Например, у нас есть прайс-листы с идентификаторами 2 и 50, то необходимо для каждого задать цены через параметры с названиями pl_2 и pl_50	CRU

Фильтры:

Название	Тип	Значение
config_id	число	Идентификатор конфигурации

## 10 Торговые автоматы

### 10.1 Команда запроса торговых автоматов

Метод: GET

Путь: /api/2.0/automat/Automat.php

Команда: ReadAutomatDetail

Параметры:

Название	Тип	Назначение
id	число	Идентификатор автомата
automat_model_id	число	Идентификатор модели автомата
automat_model_type	число	Тип торгового автомата: 0 — кофейный; 1 — снековый; 2 — хорека; 3 — газированная вода.
automat_model_name	строка	Название модели автомата
serial_number	строка	Серийный номер
key	строка	Ключ
modem_id	число	Идентификатор подключенного модема
fr_id	число	Идентификатор подключенной кассы
ext1	число	Тип устройства, подключенного к порту EXT1: 0 — ничего не подключено 1 — подключен картридер Инпас 2 — подключен картридер Сбербанк
production_date	дата	Дата производства
from_date	дата	Дата ввода в эксплуатацию
to_date	дата	Дата вывода из эксплуатации
client_id	число	Идентификатор клиента (если задано)
client_name	строка	Название клиента (если задано)
client_point_id	число	Идентификатор точка (если задана)
client_point_name	строка	Название точки (если задана)
control_billvalidator	число	Контролировать наличие купюроприёмника: 0 — отключено; 1 — включено;
control_coinchanger	число	Контролировать наличие монетоприёмника: 0 — отключено;

		1 — включено;
control_cashless	число	Контролировать наличие картридера: 0 — отключено; 1 — включено;
qr	число	Вывод qr кода: 0 — отключено; 3 — выводить qr код на экран терминала и на «Эфор» экран;

Фильтры:

Название	Тип	Значение
id	число	Идентификатор торгового автомата
automat_model_id	число	Идентификатор модели торгового автомата

## 10.2 Команда редактирования ТА

При редактировании можно задавать только те параметры, значения которых изменяются. Например, можно задать только id и fr\_id.

Метод: POST

Путь: /api/2.0/automat/Automat.php

Команда: UpdateAutomat

Параметры:

Название	Тип	Назначение
id	число	Идентификатор ТА
serial_number	строка	Серийный номер ТА
key	строка	Номер ключа ТА
modem_id	число	Идентификатор модема. Если не задан, то null
fr_id	число	Идентификатор подключенной кассы
ext1	число	Тип устройства, подключенного к порту EXT1: 0 — ничего не подключено; 1 — подключен картридер Инпас; 2 — подключен картридер Сбербанк.
production_date	дата	Дата производства ТА
from_date	дата	Дата ввода в эксплуатацию ТА
to_date	дата	Дата вывода из эксплуатации
control_billvalidator	число	Контролировать наличие купюроприёмника: 0 — отключено; 1 — включено;
control_coinchanger	число	Контролировать наличие монетоприёмника: 0 — отключено;

		1 — включено;
control_cashless	число	Контролировать наличие картридера: 0 — отключено; 1 — включено;
qr	число	Вывод qr кода: 0 — отключено; 3 — выводить qr код на экран терминала и на «Эфор» экран;

### 10.3 Команда привязки конфигурации к ТА

Метод: POST

Путь: /api/2.0/automat/Config.php

Команда: CreateAutomatConfig

Параметры:

Название	Тип	Назначение
id	число	Идентификатор связи конфигурации с ТА
automat_id	число	Идентификатор ТА
config_id	число	Идентификатор конфигурации
from_date	дата	Дата и время, с которого будет действовать конфигурация
to_date	дата	Дата и время, до которого будет действовать конфигурация. При установке должна быть равна null.

### 10.4 Команда отвязывания конфигурации от ТА

Метод: POST

Путь: /api/2.0/automat/Config.php

Команда: DeleteAutomatConfig

Параметры:

Название	Тип	Назначение
id	число	Идентификатор связи конфигурации с ТА

Примечание:

Удалять можно только самую последнюю конфигурацию. То есть удаление связей ТА с конфигурациями из середины или из начала заблокировано.

### 10.5 Команда добавления точки для ТА

Метод: POST

Путь: /api/2.0/automat/Location.php

Команда: CreateAutomatLocation

Параметры:

Название	Тип	Назначение
automat_id	число	Идентификатор ТА
automat_model_name	строка	Модель автомата (может быть пустым)
company_id	число	Идентификатор контрагента
company_name	число	Название контрагента (может быть пустым)
company_point_id	число	Идентификатор точки контрагента
company_point_name	строка	Название точки контрагента (может быть пустым)
from_date	дата	Дата и время, с которого будет действовать точка
to_date	дата	Дата и время, до которого будет действовать точка. При установке должна быть равна null.

## 10.6 Команда чтения точек для ТА

Метод: GET

Путь: /api/2.1/automat/Location.php

Команда: ReadAutomatLocationDetail

Фильтры:

Название	Тип	Значение
automat_id	число	Смотрите одноименный параметр
company_point_id	строка	Смотрите одноименный параметр
to_date	число	Смотрите одноименный параметр

## 10.7 Команда отвязывания точки от ТА

Метод: POST

Путь: /api/2.1/automat/Location.php

Команда: DeleteAutomatLocation

Параметры:

Название	Тип	Назначение
id	число	Идентификатор точки

Примечание:

Удалять можно только самую точку. То есть удаление связей ТА с точками из середины или из начала заблокировано.

## 10.8 Команда изменения точки

Метод: POST

Путь: /api/2.1/automat/Location.php

Команда: UpdateAutomatLocation

Параметры:

Название	Тип	Назначение
id	число	Идентификатор точки
to_date	дата	Дата и время, до которого будет действовать точка.

## 11 Устройства автомата

### 11.1 Команда запроса одного устройства

Метод: GET

Путь: api/2.0/automat/Device.php

Команда: ReadAutomatOneDevice

Параметры:

Название	Тип	Назначение
id	число	Идентификатор устройства
type	число	Тип устройства: 1 — модем; 2 — автомат; 3 — касса; 4 — купюроприемник; 5 — монетоприёмник; 6 — картридер 1; 7 — картридер 2; 8 — картридер EXT1; 9 — экран; 10 — NFS считыватель; 11 — картридер USB1.
type_name	строка	Смотри поле type
state	строка	Состояние устройства: 0 — не найдено; 1 — инициализация; 2 — работает; 3 — ошибка; 4 — выключен; 5 — включен.
state_name	строка	Смотри поле state

Фильтры:

Название	Тип	Назначение
type	число	Тип устройства.
automat_id	число	Идентификатор автомата.

## 11.2 Команда запроса информации о устройстве

Метод: GET

Путь: api/2.0/automat/Device.php

Команда: ReadAutomatDeviceParam

Параметры:

Название	Тип	Назначение
id	число	Идентификатор параметра устройства
device_id	число	Идентификатор устройства
type	строка	Тип параметра устройства
type_name	строка	Название типа устройства
value	строка	Значение параметра устройства

Фильтры:

Название	Тип	Назначение
device_id	число	.Идентификатор устройства

## 12 Аудиты торговых автоматов

Учет и контроль продаж построен на основе срезов счетчиков продаж торгового автомата. Здесь и далее срез счетчиков продаж торгового автомата будем называть «аудитом».

В системе предусмотрено два контура учета:

1. Учет наличных;
2. Учет продаж по факту продажи.

### 12.1 Учет наличных

Для учета наличных используются аудиты типа 0 и 1.

Учет наличных организован следующим образом: в момент инкассации снимается аудит и регистрируется номер мешка. При подсчете мешка в систему заносится фактически поступившая сумма наличных. Также операторы задают в систему суммы, на которые они пополняют размен в торговых автоматах.

При добавлении аудита в систему или при обновлении данных в пределах действия аудита (то есть от даты регистрации текущего аудита до даты регистрации предыдущего аудита) аудит пересчитывается по следующей формуле:

$$\text{encash\_val\_conflict} = \text{vend\_cash\_val\_dif} + \text{tube\_in\_val\_manual} - \text{tube\_val\_dif} - \text{encash\_val\_manual}$$

где

vend\_cash\_val\_dif — расчетная сумма продаж в рублях. Рассчитывается как разница несбрасываемого счетчика продаж в рублях текущего аудита и предыдущего аудита;

tube\_in\_val\_manual — пополнение нала в тубах операторами;

tube\_val\_dif — изменения нала в тубах. Рассчитывается как разница нала в тубах текущего аудита и предыдущего аудита;

encash\_val\_manual — сумма фактически инкассированных оператором наличных за период действия аудита;

encash\_val\_conflict — результат контроля наличных. 0 — идеальная ситуация, недостаток не обнаружено. Положительное число — недостача. Отрицательное число — избыток.

Если необходимо по какой-то причине начать учет наличных заново, то аудиту нужно задать тип «Точка отсчета». Аудит с типом «Точка отсчета» устанавливает все счетчики на ноль и не будет обновляться при обновлении данных в период его действия.

## 12.2 Учет продаж по факту продажи

Для учета продаж по факту продажи используются аудиты с типом 3.

Данный аудит создается автоматически для каждого автомата каждый день и пополняется данными о продажах по мере поступления от модемов информации о продажах за нал или безнал. Если для какого-то автомата данного аудита в какой-то день нет, то в систему не поступило информации ни об одной продаже в этот день в этом автомате.

## 12.3 Таблица аудитов

Путь: /api/2.0/audit/Audit.php

Команды: ReadAudit, DeleteAudit

Параметры:

Название	Тип	Назначение	Применение
id	число	Идентификатор записи	RD
automat_id	число	Идентификатор автомата. Он же сейчас является номером автомата.	R
serial_number	строка	Серийный номер автомата	R
operator_id	число	Идентификатор пользователя, который снял аудит через телефон. Если аудит снят модемом, что будет равно NULL.	R
date	дата	Дата и время, когда был снят аудит	R
type	число	Тип аудита: 0 — аудит снятый при инкассации; 1 — аудит точка отсчета; 3 — автоматический аудит;	R
method	число	Способ снятия аудита: 0 — введен вручную через веб-клиент;	R

		1 — введен вручную через телефон; 2 — загружен через телефон; 3 — импортирован в формате EVADTS (как правило снять с помощью флэшки) 4 — снят с помощью модема; 5 — создан сервером.	
vend_cash_num	число	Продажи за нал в штуках	R
vend_cash_num_dif	число	Разница продаж за нал в штуках. Разница между vend_cash_num текущего аудита и предыдущего, для все остальных параметров применяется аналогичный принцип.	R
vend_cash_val	число	Продажи за нал в рублях. Цены указаны в рублях без копеек. Другими словами двоичная точка равна 0.	R
vend_cash_val_dif	число	Разница продаж за нал рублями.	R
encash_val_manual	число	Инкассировано средств в рублях за период с момента снятия предыдущего аудита по дату снятия текущего аудита включительно.	R
encash_val_conflict	число	Отклонение расчетной суммы от фактически инкассированной. Положительное значение показывает недостачу. Отрицательное значение — избыток.	R
tube_in_val	число	Зарезервировано	R
tube_in_val_dif	число	Зарезервировано	R
tube_in_val_manual	число	Сумма пополнения нала в тубах за период с момента снятия предыдущего аудита по дату снятия текущего аудита включительно.	R
tube_in_val_conflict	число	Зарезервировано	R
tube_val	число	Сумма нала в тубах в момент снятия аудита.	R
tube_val_dif	число	Разница нала в тубах	R
vend_cashless_num	число	Продажи за безнал в штуках	R
vend_cashless_num_dif	число	Разница продаж за безнал в штуках	R
vend_cashless_val	число	Продажи за безнал в рублях	R
vend_cashless_val_dif	число	Разница продаж за безнал в рублях	R

## Фильтры:

Название	Тип	Значение
automat_id	число	Фильтр по идентификатору автомата
operator_id	число	Фильтр по идентификатору оператора
type	число	Фильтр по типу
method	число	Фильтр по методу

date_from date_to	дата	Фильтр по диапазону (параметр «date»). Должны одновременно задаваться левая и правая границы диапазона.
----------------------	------	---

## 12.4 Счетчики продаж по продуктам

Метод: GET

Путь: /api/2.0/audit/Audit.php

Команда: ReadAuditProductDetail

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
audit_id	число	Идентификатор аудита
select_id	строка	Идентификатор продукта
ware_id	число	ID продукта в системе
ware_code	строка	Артикул продукта
name	строка	Название продукта
vend_num	число	Продажи в штуках
vend_num_dif	число	Разница продаж в штуках
vend_val	число	Продажи в рублях
vend_val_dif	число	Разница продаж в рублях
vend_cash_num	число	Продажи за нал в штуках
vend_cash_num_dif	число	Разница продаж за нал в штуках
vend_cash_val	число	Продажи за нал в рублях
vend_cash_val_dif	число	Разница продаж за нал в рублях
vend_cashless_num	число	Продажи за безнал в штуках
vend_cashless_num_dif	число	Разница продаж за безнал в штуках
vend_cashless_val	число	Продажи за безнал в рублях
vend_cashless_val_dif	число	Разница продаж за безнал в рублях
<PLID>_price	число	Цена данного продукта в заданном префиксом прайс-листе. <PLID> - идентификатор прайс листа. Например, «CA0», «DA1».
<PLID>_vend_num	число	Продажи по заданному прайс-листу в штуках
<PLID>_vend_num_dif	число	Разница продаж по заданному прайс-листу в штуках
<PLID>_vend_val	число	Продажи по заданному прайс-листу в рублях
<PLID>_vend_val_dif	число	Разница продаж по заданному прайс-листу в рублях

Фильтры:

Название	Тип	Значение
audit_id	число	Идентификатор аудита

Примечание:

В автоматических аудитах (тип 3) не возвращается детализация по прайс-листам (параметры <PLID>\_price, <PLID>\_vend\_num, <PLID>\_vend\_num\_dif, <PLID>\_vend\_val, <PLID>\_vend\_val\_dif).

## 12.5 Аудиты со счетчиками продаж по продуктам

Путь: /api/2.0/audit/Audit.php

Команды: ReadAuditFull

Ограничение 25 записей за один запрос.

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
automat_id	число	Идентификатор автомата. Он же сейчас является номером автомата.
operator_id	число	Идентификатор пользователя, который снял аудит через телефон. Если аудит снят модемом, что будет равно NULL.
date	дата	Дата и время, когда был снят аудит
type	число	Тип аудита: 0 — аудит снятый при инкассации; 1 — аудит точка отсчета; 3 — автоматический аудит;
method	число	Способ снятия аудита: 0 — введен вручную через веб-клиент; 1 — введен вручную через телефон; 2 — загружен через телефон; 3 — импортирован в формате EVADTS (как правило снять с помощью флэшки) 4 — снят с помощью модема; 5 — создан сервером.
vend_cash_num	число	Продажи за нал в штуках
vend_cash_num_dif	число	Разница продаж за нал в штуках. Разница между vend_cash_num текущего аудита и предыдущего, для все остальных параметров применяется аналогичный принцип.
vend_cash_val	число	Продажи за нал в рублях. Цены указаны в рублях без копеек. Другими словами двоичная точка равна 0.
vend_cash_val_dif	число	Разница продаж за нал рублях.
encash_val_manual	число	Инкассировано средств в рублях за период с момента снятия предыдущего аудита по дату снятия текущего аудита включительно.
encash_val_conflict	число	Отклонение расчетной суммы от фактически инкассированной. Положительное значение показывает недостачу. Отрицательное значение — избыток.
tube_in_val	число	Зарезервировано
tube_in_val_dif	число	Зарезервировано
tube_in_val_manual	число	Сумма пополнения нала в тубах за период с момента

		снятия предыдущего аудита по дате снятия текущего аудита включительно.
tube_in_val_conflict	число	Зарезервировано
tube_val	число	Сумма нала в тубах в момент снятия аудита.
tube_val_dif	число	Разница нала в тубах
vend_cashless_num	число	Продажи за безнал в штуках
vend_cashless_num_dif	число	Разница продаж за безнал в штуках
vend_cashless_val	число	Продажи за безнал в рублях
vend_cashless_val_dif	число	Разница продаж за безнал в рублях
audit_product	массив	Массив счётчиков продаж по продуктам

Параметры элемента массива audit\_product:

Название	Тип	Значение
id	число	Идентификатор записи
audit_id	число	Идентификатор аудита
select_id	строка	Идентификатор продукта
name	строка	Название продукта
vend_num	число	Продажи в штуках
vend_num_dif	число	Разница продаж в штуках
vend_val	число	Продажи в рублях
vend_val_dif	число	Разница продаж в рублях
vend_cash_num	число	Продажи за нал в штуках
vend_cash_num_dif	число	Разница продаж за нал в штуках
vend_cash_val	число	Продажи за нал в рублях
vend_cash_val_dif	число	Разница продаж за нал в рублях
vend_cashless_num	число	Продажи за безнал в штуках
vend_cashless_num_dif	число	Разница продаж за безнал в штуках
vend_cashless_val	число	Продажи за безнал в рублях
vend_cashless_val_dif	число	Разница продаж за безнал в рублях
<PLID>_price	число	Цена данного продукта в заданном префиксом прайс-листе. <PLID> - идентификатор прайс листа. Например, «CA0», «DA1».
<PLID>_vend_num	число	Продажи по заданному прайс-листу в штуках
<PLID>_vend_num_dif	число	Разница продаж по заданному прайс-листу в штуках
<PLID>_vend_val	число	Продажи по заданному прайс-листу в рублях
<PLID>_vend_val_dif	число	Разница продаж по заданному прайс-листу в рублях

Фильтры:

Название	Тип	Значение
automat_id	число	Фильтр по идентификатору автомата
operator_id	число	Фильтр по идентификатору оператора
type	число	Фильтр по типу
method	число	Фильтр по методу
date_from date_to	дата	Фильтр по диапазону (параметр «date»). Должны одновременно задаваться левая и правая границы диапазона.

## 12.6 Аудит в формате EVADTS

Путь: /api/2.0/audit/Audit.php

Команды: ReadTextAudit

Фильтры:

Название	Тип	Значение
id	число	Фильтр по идентификатору аудита

## 11.7 Аудит статистики

Тип: 2 обозначает статистический аудит

Тип 4 обозначает точку отчёта статистического аудита.

Данный аудит не участвует в учёте и контроле денежных средств.

## 13 Загрузки торговых автоматов

Получение информации о загрузке вместе со значениями товаров.

### 13.1 Загрузки

Ограничение 100 записей за один запрос.

Обратите внимание что если при завершении загрузки в течении двух часов с создания «Аудита» который формируется при нажатии кнопки 1 на модеме прошло менее 2-х часов, то дата операции изменится на дату ближайшего прошедшего аудита (тем самым загрузка оказывается привязана к аудиту и его счётчики можно использовать для проверки правильности операции).

Метод: GET

Путь: /api/2.0/ware/WareFlow.php

Команда: ReadWareFlowFull

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
date	дата	Дата и время сборки (либо дата последнего аудита)
automat_id	число	Номер автомата
collector_id	число	Идентификатор пользователя создавшего сущность
collector_name	число	Фамилия и Имя пользователя создавшего сущность
operator_id	число	Идентификатор пользователя осуществившего загрузку\выгрузку
operator_name	число	Фамилия и Имя пользователя осуществившего загрузку\выгрузку
state	число	Состояние операции: 0 – Создана 1 – Завершена
type	число	Тип операции: 0 – Сборка/Загрузка 1 – Выгрузка
products	массив	Продукты и их значения

Параметры элемента массива products:

Название	Тип	Значение
id	число	Идентификатор записи
wareflow_id	число	Идентификатор операции
select_id	число	Код ячейки
ware_id	число	Код продукта из Номенклатуры

qt_max	число	Максимальное количество товара в ячейке
qt_recom	число	Рекомендуемое количество товара в ячейке
qt_rest	число	Остаток в автомате на момент создания записи
qt_rest_fact	число	Остаток введённый оператором при загрузке
qt_take	число	Рекомендуемое количество товара для сборки
qt_take_fact	число	Фактическое количество товара взятого для сборки
qt_fill	число	Количество товара которое должно быть загружено
qt_fill_fact	число	Количество товара которое было загружено
qt_pull	число	Количество выгруженного товара
qt_counter	число	Счётчик текущего количества товара в ячейке, реальное количество товара всегда всегда находится в самой последней завершённой загрузке - (state = 1)

Фильтры:

Название	Тип	Значение
automat_id	число	Номер автомата
type	число	Смотрите одноимённый параметр
date_from date_to	дата	Фильтр по диапазону (параметр «date»). Должны одновременно задаваться левая и правая границы диапазона.

## 14 Команды модему

Команды, которые можно отправить модему через личный кабинет

### 14.1 Команды модему

Метод: POST

Путь: /api/2.0/Command.php

Команда: Create

Параметры:

Название	Тип	Назначение
modem_id	число	Идентификатор модема
command	число	Команда модему: 1 - Загрузка настроек 2 - Перезагрузка модема 3 - Перезагрузка автомата 4 - Сброс ошибок 5 - Начисление кредита 6 - Выгрузка аудита 7 - Обновление прошивки 8 - Сверка итогов 9 - Сброс счетчика аудита 10 - Создание статического аудита

+

## 15 Проведение платежа через внешнее приложение

### 15.1 Создание транзакции продажи

Метод: POST

Путь: /api/2.0/PaymentEphorPay.php

Команда: PaymentToken

Параметры:

Название	Тип	Назначение	Обязательность
pay_type*	число	Сценарии оплаты: 1 – блокировка денежных средств с предвыбором товара 2 — блокировка денежных средств равная цене самого дорогого товара в ассортименте, с последующим снятием стоимости выбранного товара на автомате и	Да

		разблокировкой разницы	
token	строка	Идентификатор транзакции	
account_id	число	Идентификатор аккаунта	Да
automat_id	число	Идентификатор автомата	Да
token_type	число	Тип токена: 1 – Google Pay 2 – Apple Pay 3 – Samsung Pay	Нет
device_type	число	0 — кофейный автомат; 1 — снековый автомат; 2 — хорека-автомат; 3 — автомат газированной воды; 4 — механический автомат; 5 — комбинированный автомат; 7 — холодильник	
sum	число	Сумма платежа в минимальных единицах валюты	
qr_format	число	1 — картинка в формате base64 2 — url-ссылка в формате строки	

Параметры в ответе:

Название	Тип	Назначение
data	число	Идентификатор транзакции

## 15.2 Получения состояния транзакции

Метод: POST

Путь: /api/2.0/PaymentEphorPay .php

Команда: GetStatusPaymentToken

Параметры:

Название	Тип	Назначение
pay_id	строка	Идентификатор транзакции
account_id	число	Идентификатор аккаунта

Параметры в ответе:

Название	Тип	Назначение
status	число	Состояние транзакции: 1 - пользователь отправил токен для инициализации покупки;

		2 - ожидание на оплату, на подтверждения от автомата и тд; 3 - пользователь должен нажать кнопку на автомате; 4 - пользователь получил свой товар; 5 - пользователь не смог получить свой товар по любой причине. Причина указана в поле error; 6 - пользователю возвращены деньги;
description	строка	Текстовое описание текущего состояния
error_code	число	Детализация ошибки: 0x0201 - ошибка блокировки денег 0x0202 - ошибка списания денежных средств 0x0301 - ошибка выдачи продукта 0x0302 - сессия оплаты отменена автоматом 0x0303 - сессия оплаты 0x0304 - выбран не тот продукт, который предоплачен 0x0305 - выдача товара отменена автоматом
qr_format	число	1 — картинка в формате base64 2 — url-ссылка в формате строки
f_qr	строка	url-ссылка в формате строки или картинка в формате base64 чека

## 16 Отчеты

### 16.1 Отчет по товарам

Отчет формируется строго по целым дням. Если задать время, то результат будет некорректный.

В отчете один и тот же товар будет выведен в разных строках, если товар продается по разным ставкам налога.

Если для продукт в конфигурации не был связан с товаром из списка номенклатуры, то такой товар будет выведен отдельной строкой и будет иметь значение ware\_id равным NULL. То есть если у продукта, связанного с товаром из списка номенклатуры, будет совпадать название с продуктом, несвязанным с товаром из списка номенклатуры, то объединение не произойдет и такие продажи будут показаны разными строчками.

Метод: GET

Путь: /api/2.0/report/sale/product/json.php

Команда: Read

Параметры:

Название	Тип	Назначение
ware_id	число	Идентификатор товара из списка номенклатуры
ware_code	число	Артикул товара из списка номенклатуры

name	строка	Название продукта
total_num	число	Общее количество продаж
total_val	число	Общая сумма продаж
vend_cash_num	число	Продажи за нал в штуках
vend_cash_val	число	Продажи за нал в рублях
vend_cashless_num	число	Продажи за безнал в штуках
vend_cashless_val	число	Продажи за безнал в рублях
vend_num	число	Итого продаж в штуках
vend_val	число	Итого продаж в рублях
tax_rate	число	Ставка налога
tax_value	число	Сумма налога к уплате по заданной ставке

Фильтры:

Название	Тип	Значение
date_from	дата	Дата, начиная с которой формировать отчет. Время в этой дате должно быть 00:00:00
date_to	дата	Дата, по которую включительно формировать отчет. Время в этой дате должно быть 23:59:59
automat_id	число	Идентификатор торгового автомата. В отчет будут включены только продажи заданного торгового автомата.
fiscalized	число	Фильтр не обязателен. 0 - фискализация отключена. 1 - фискализация успешна. 2 - ошибка фискализации.
status	число	Фильтр не обязателен. 0 - продажа за 0 рублей или касса отключена. 1 - чек создан успешно, реквизиты получены. 2 - чек добавлен в очередь, реквизиты не получены. 3 - результат постановки чека в очередь не известен. 4 - ошибка создания чека. 5 - очередь отложенной регистрации переполнена. 6 - чек фискализирован вручную.  *Обратите внимание что при одновременном использовании фильтров fiscalized и status — фильтрация произойдет только по фильтру status.

## 16.2 Отчет продаж от аудита до заданной даты

Порядок формирования отчета:

1. Ищется до заданной даты ближайший к заданной дате аудит;
2. Выбираются все учтенные продажи между датой аудита и заданной датой;

3. Корректируются счетчики продаж по кнопкам в соответствии с обнаруженными продажами.

Метод: GET

Путь: /api/2.0/report/sale/encashment/json.php

Команда: Read

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
audit_id	число	Идентификатор аудита
select_id	строка	Идентификатор продукта
name	строка	Название продукта
vend_cash_num	число	Продажи за нал в штуках
vend_cash_num_dif	число	Разница продаж за нал в штуках
vend_cash_val	число	Продажи за нал в рублях
vend_cash_val_dif	число	Разница продаж за нал в рублях
vend_cashless_num	число	Продажи за безнал в штуках
vend_cashless_num_dif	число	Разница продаж за безнал в штуках
vend_cashless_val	число	Продажи за безнал в рублях
vend_cashless_val_dif	число	Разница продаж за безнал в рублях

Фильтры:

Название	Тип	Значение
automat_id	число	Идентификатор автомата
date	дата	Дата, до которой ищется аудита

## 16.3 Отчет по событиям заданного автомата

Метод: GET

Путь: /api/2.0/report/automat/Event.php

Команда: Read

Параметры:

Название	Тип	Назначение
event_id	число	Идентификатор записи
event_type	число	Тип события: 0 – событие; 1 – продажа; 2 – аудит; 3 – инкассация или размен.
date	дата	Дата и время возникновения события
name	строка	Название события
description	строка	Описание
fiscalized	число	Может отсутствовать в ответе. 0 – не фискализировано (без чека) 1 – фискализировано (с чеком)

Фильтры:

Название	Тип	Значение
automat_id	число	Фильтр обязателен. Идентификатор автомата
category	массив	Фильтр не обязателен. 0 — Ошибка. 1 — Продажа. 2 — Информация. 3 — Движение денег. 4 — Задачи. (не реализовано) 5 — Приём денег.
date_from date_to	дата	Фильтр не обязателен. Фильтр по диапазону (параметр «date»). Должны одновременно задаваться левая и правая границы диапазона.

## 16.4 Отчет по списку чеков

Метод: GET

Путь: /api/2.0/report/sale/Sale.php

Команда: Read

## Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
date	дата	Дата и время возникновения события
automat_id	число	Идентификатор автомата
select_id	число	Номер ячейки
ware_id	число	Идентификатор записи в номенклатуре
ware_code	строка	Артикул
payment_device	число	Метод оплаты: 1 – наличные; 2 – безналичные.
name	строка	Название продукта
credit	число	Сумма внесённая клиентом
value	число	Цена товара в рублях
tax_system	число	Система налогообложения: 0x00 — функционал в реализации; 0x01 — Общая ОСН; 0x02 — Упрощенная доход; 0x04 — Упрощенная доход минус расход; 0x08 — Единый налог на вмененный доход; 0x10 — Единый сельскохозяйственный налог; 0x20 — Патентная система налогообложения.
tax_rate	число	Налоговая ставка: 0 — без НДС; 1 — НДС 0%; 2 — НДС 10%; 3 — НДС 20%. 4 — НДС 12%. 5 — НДС 5%. 6 — НДС 7%. 7 — НДС 5/105. 8 — НДС 7/107. 9 — НДС 10/110. 10 — НДС 20/120. 11 — НДС 22%. 12 — НДС 22/122.
tax_value	число	Сумма налога в копейках
fn	число	Номер фискального накопителя
fd	число	Номер фискального документа
fp	число	Фискальный признак
status	число	0 - продажа за 0 рублей или касса отключена. 1 - чек создан успешно, реквизиты получены.

		2 - чек добавлен в очередь, реквизиты не получены. 3 - результат постановки чека в очередь не известен. 4 - ошибка создания чека. 5 - очередь отложенной регистрации переполнена. 6 - чек фискализирован вручную.
--	--	---

Фильтры:

Название	Тип	Значение
automat_id	число	Фильтр не обязателен. Идентификатор автомата
id	число	Id чека в базе «Эфор». Выбираются чеки с id и дальше ( $\geq$ ).
fiscalized	число	Фильтр не обязателен. 0 - фискализация отключена. 1 - фискализация успешна. 2 - ошибка фискализации.
status	массив	Фильтр не обязателен. 0 - продажа за 0 рублей или касса отключена. 1 - чек создан успешно, реквизиты получены. 2 - чек добавлен в очередь, реквизиты не получены. 3 - результат постановки чека в очередь не известен. 4 - ошибка создания чека. 5 - очередь отложенной регистрации переполнена. 6 - чек фискализирован вручную.  *Обратите внимание что при одновременном использовании фильтров fiscalized и status — фильтрация произойдёт только по фильтру status.
date_from date_to	дата	Фильтр не обязателен. Фильтр по диапазону (параметр «date»). Должны одновременно задаваться левая и правая границы диапазона

Примечания:

- **Лимит на кол-во записей, которые отдает сервер, 20000.**
- **Для страничного получения данных необходимо указывать offset и limit. Offset – значение указывает с какой записи выбирать данные, а Limit указывает максимальное количество записей. Offset  $\geq$  кол-во записей  $\leq$  Limit.**
- **Если в запросе передан id чека из базы, то все остальные фильтры будут игнорироваться.**

## 16.5 Получение списка модемов

Метод: GET

Путь: /api/2.0/Account.php

Команда: ReadAccountModem

Параметры:

Название	Тип	Назначение
id	число	Идентификатор записи
imei	строка	IMEI модема
hardware_version	число	Версия платы модема Состоит из 4 байт: байт 3 — первое число байт 2 — второе число байты 1 и 0 — третье число
software_version	число	Версия прошивки модема Состоит из 4 байт: байт 3 — первое число байты 2 — второе число байт 1 и 0 — третье число
software_release	число	Релиз прошивки: 0x00 - Не обновлять; 0x01 - Стабильный 0x02 - Новый 0x81 - Техподдержка1 0x82 - Техподдержка2 0x83 - Техподдержка3 0x84 - Техподдержка4 0x85 - Техподдержка5
automat_id	число	Идентификатор записи автомата. Может быть равен NULL, если модем не связан с автоматом.
signal_quality	число	Уровень сигнала
last_login	дата	Дата и время последнего подключения модема
phone	строка	Не используется
static	число	Статический IP: 0 – отключен 1 — включен
gsm_apn	строка	Идентификатор сети, Может быть равен NULL
gsm_username	строка	пользователь сети, Может быть равен NULL
gsm_password	строка	Пароль пользователя сети, Может быть равен NULL
dns1	строка	ДНС адрес, Может быть равен NULL
dns2	строка	Дополнительный ДНС адрес, Может быть равен NULL

Фильтры:

Название	Тип	Значение
id	число	Фильтр не обязателен. Идентификатор модема
automat_id	число	Фильтр не обязателен. Идентификатор автомата
imei	строка	Фильтр не обязателен. IMEI модема. Может быть задан фрагмент

## 16.6 Отчёт «Состояние торговых автоматов»

Метод: GET

Путь: /api/2.0/report/automat/State.php

Команда: Read

Параметры:

Название	Тип	Назначение
automat_id	число	Номер автомата
model_id	строка	Идентификатор модели автомата
model_name	строка	Название модели автомата
model_type	число	Тип модели автомата: 0 — Кофейный 1 — Снековый 2 — Хорека 3 — Газ вода 4 — Механический 5 — Комбинированный
modem_signal	число	Уровень сигнала у модема, число от 0 до 100.
last_login	дата	Дата последнего подключения модема
modem_id	число	Идентификатор модема
notification_id	число	Идентификатор шаблона оповещений
now_cash_val	число	Сумма продаж с момента последнего аудита за наличный расчёт
now_cash_num	число	Количество продаж с момента последнего аудита за наличный расчёт
now_cashless_num	число	Количество продаж с момента последнего аудита за безналичный расчёт
num	число	Количество продаж с момента последнего аудита (общее)
tube_val	число	Количество размена в автомате с момента последнего аудита
now_tube_val	число	Текущее количество размена в автомате

modem_state	число	Состояние модема: 0 — Не подключен 1 — Нет связи 2 — На связи
automat_state	число	Состояние автомата: 0 — Ошибка 1 — Предупреждение 2 — Нет данных 3 — Работает
point_id	число	Идентификатор точки
point_name	строка	Название точки
company_id	число	Идентификатор контрагента
company_name	строка	Название контрагента
point_adress	строка	Адрес точки
config_id	число	Идентификатор активной конфигурации
config_name	строка	Название активной конфигурации
type_service_time	число	Данные для подсветки поля «Аудит»: 0 — по графику 1 — допустимое нарушение периодичности обслуживания 2 — недопустимое нарушение периодичности обслуживания
type_encashment_time	число	Данные для подсветки поля «Инкассация»: 0 — по графику 1 — допустимое нарушение периодичности инкассации 2 — недопустимое нарушение периодичности инкассации
type_nosale_time	число	Данные для подсветки поля «Продажа»: 0 — плановые продажи 1 — продаж нет слишком долго 2 — недопустимое время нет продаж
audit_date	дата	Дата последнего обслуживания
encash_date	дата	Дата последней инкассации
sale_date	дата	Дата последней продажи
date	дата	Текущая дата и время
tube_balance	число	Количество размена (из устройства Монетоприёмник)
type_tube_balance	число	Данные для подсветки поля «Размен»: 0 — размена достаточно 1 — допустимое количество размена 2 — недопустимое количество размена
encash_val	число	Сумма к инкассации
deviceN	число	DeviceN – где N это тип устройства: 1 — Модем

		2 — Автомат 3 — Фискальный регистратор 4 — Купюроприёмник 5 — Монетоприёмник 6 — Картридер 1 7 — Картридер 2 8 — Дополнительный Картридер 9 — Эфор экран 10 — Модуль бесконтактной оплаты  Статусы устройств: 0 — Устройство не найдено 1 — Инициализация 2 — Работает 3 — Ошибка 4 — Устройство отключено 5 — Устройство подключено
--	--	--

Фильтры:

Название	Тип	Значение
id	число	Фильтр не обязателен. Идентификатор модема
automat_id	число	Фильтр не обязателен. Идентификатор автомата
modem_devices	массив	Фильтр не обязателен. Активные типы устройств: 3 — Фискальный регистратор 4 — Купюроприёмник 5 — Монетоприёмник 6 — Картридер 10 — Модуль бесконтактной оплаты
client_id	число	Фильтр не обязателен. Идентификатор клиента
modem_state	массив	Фильтр не обязателен. Состояние модема: 0 — Не подключен 1 — Нет связи 2 — На связи
automat_model_id	число	Фильтр не обязателен. Идентификатор модели автомата
type	массив	Фильтр не обязателен. Тип модели автомата: 0 — Кофейный 1 — Снековый

Эфор. Решения телеметрии и фискализации.

		2 — Хорека 3 — Газ вода 4 — Механический 5 — Комбинированный
--	--	---