

	ТР _____	
--	----------	--

Технический Регламент № \_\_\_\_\_

ДАТА ВЫПУСКА ДОКУМЕНТА  
11.09.2017

ДОКУМЕНТ РАЗРАБОТАН

ФИО	Должность	Дата
Григорьев Д.С.	Главный специалист, Отдел информатизации и развития бизнес-процессов	11.09.2017
Луценко Д.В.	Главный специалист, Отдел информатизации и развития бизнес-процессов	01.05.2018

СОДЕРЖАНИЕ ДОКУМЕНТА

ФИО	Документ	Дата

ДОКУМЕНТ СОГЛАСОВАН

ФИО	Должность	Дата
Прокофьев В.С.	Менеджер, Отдел информатизации и развития бизнес-процессов	11.09.2017
Прокофьев В.С.	Менеджер, Отдел информатизации и развития бизнес-процессов	01.05.2018

ОТВЕТСТВЕННЫЙ ЗА ПРОЕКТ

ФИО	Должность	Дата
Прокофьев В.С.	Менеджер, Отдел информатизации и развития бизнес-процессов	11.09.2017

	2.0.0.32	1 / 65
--	----------	--------

	ТР _____	
--	----------	--

## ВЕДОМОСТЬ ИЗМЕНЕНИЙ

Дата изменения	ИЗМЕНЕНИЯ тип (дополнения, удаления) и конкретное описание
01.09.2017	Создание документа
23.10.2017	Изменение документа (содержание; добавление категории товара; описание метода, возвращающего статус заявки).
25.12.2017	Исправление опечатки contractCode на contract
	Добавлен метод получения кода договора по карте.
18.01.2018	Поправлена опечатка в лимитах. Замена поля Code на LimitCode. Добавлены примеры на javascript. Удалено поле RestrictionCode, использовать только RestrictionId.
05.04.2018	Работа с недельными лимитами
03.05.2018	Исправлено описание возвращаемого результата в методах 2.4.4 и 2.4.5
17.05.2018	Исправлены неточности в описании товарных ограничителей
21.05.2018	Добавлен метод 2.7.1 – «Получение списка групп карт»
21.05.2018	Добавлен метод 2.7.2 – «Управление группами карт»
21.05.2018	Добавлен метод 2.7.3 – «Управление картами в группах»
21.05.2018	Добавлены V2 методы получения транзакций (информация о скидке и идентификаторе АЗС)
25.05.2018	Метод 2.6.3 обновился до версии V2. Параметр card является строковым значением.
25.05.2018	Метод 2.7.1. В возвращаемом ответе, номер карты является строковым значением.
25.05.2018	В методе 2.7.3 параметр CardList[] должен являться массивом строк.
28.05.2018	Исправлен пример вывода в XML формате для метода 2.5.1
18.06.2018	Исправлены неточности в описании методов блокировки/разблокировки карт
11.07.2018	Добавлен период лимита - квартал.
10.08.2018	Добавлен метод 2.4.6 – «Получение информации о статусе карт»
13.08.2018	Добавлен метод 2.4.7 – «Количество попыток ввода пин-кода карты»
17.08.2018	Исправлена опечатка в имени метода 2.4.3.3 «Удаление товарного ограничителя»
17.09.2018	Добавлен пропущенный параметр PosCode в примере вывода результата методов 2.5.1 и 2.5.2
04.12.2018	Добавлен новый код ошибки
20.12.2018	Создание общего лимита
20.12.2018	MD5-хэш пароля
28.02.2019	Пароль можно передавать в заголовке HTTP-запроса
12.03.2019	Добавлены новые коды ошибок и пункт 1.4 про белый список IP-адресов
21.02.2020	Добавлен новый метод 2.4.8 «Закрытие виртуальной карты»
08.09.2020	Добавлен метод 2.3.2 «Региональные ограничители договора»
08.09.2020	Добавлен метод 2.4.9 «Региональные ограничители карты»
08.09.2020	Добавлен метод 2.4.2.5 «Удаление всех лимитов карты»
08.09.2020	Добавлен метод 2.4.3.4 «Удаление всех товарных ограничителей карты»
19.10.2020	Добавлен метод 2.6.4 «Получение информации о странах и их регионах»
06.05.2021	Добавлены методы: 2.6.5 «Получение списка партнёров» 2.6.6 «Получение списка НПО» 2.6.7 «Получение списка ТО» Добавлена возможность установки региональных ограничителей карт и договоров по партнёрам и НПО. Изменены методы получения региональных ограничителей карт и договоров: числовые идентификаторы партнёра и НПО заменены на коды. Добавлено описание работы с региональными ограничителями в п. 2.3.2.
06.07.2021	Добавлен метод 2.5.3 «Получение списка операций по договору по дате изменения»
30.09.2021	Исправлена опечатка в адресе метода 2.4.8 Закрытие виртуальной карты
26.10.2021	Добавлен дополнительный параметр «GoodsInfo» в метод 2.6.7 «Получение списка ТО», позволяющий получать цены на товары/услугу и список доступных товаров/услуг на ТО. Добавлен дополнительный параметр «RType» в метод 2.4.3.2 «Создание товарного ограничителя карты», позволяющий указывать тип товарного ограничителя (запрещающий/разрешающий)

	ТР _____	
--	----------	--

Дата изменения	ИЗМЕНЕНИЯ тип (дополнения, удаления) и конкретное описание
06.12.2021	Добавлен метод 2.4.3.5 «Изменение товарного ограничителя». Добавлены дополнительные параметры GCode, GFlag, RType в метод 2.4.3.3 «Удаление товарного ограничителя», позволяющие сохранять значения в журнале изменений товарного ограничения
11.01.2022	Методы получения операций 2.5.1, 2.5.2, 2.5.3 возвращают НДС для операций обслуживания
20.01.2022	Метод получения списка ТО 2.6.7 возвращает разрешенные типы карт (топливные карты Роснефть, виртуальные карты, онлайн-оплата из мобильного приложения РН-Карт, пластиковая карта в мобильном приложении РН-Карт итд)
31.03.2022	Исправлены опечатки в методах 2.3.2.2, 2.4.2.1, 2.4.2.2, 2.4.2.3, 2.6.2, 2.6.7, 2.7.1
24.11.2022	Добавлен метод 2.6.8 «Получение информации о лимитах и/или ограничителях всех карт договора, одной карты и/или договора»
20.12.2022	Добавлен метод 2.3.3 «Получение списка ограничителей по времени для договора»
20.12.2022	Добавлен метод 2.4.10.1 «Получение списка ограничителей по времени для карты»
20.12.2022	Изменён метод 2.6.8 «Получение информации о лимитах и/или ограничителях всех карт договора, одной карты и/или договора». Теперь данный метод возвращает ещё и ограничители по времени для договора/карт.
20.12.2022	В описание метода 2.4.2.5 «Удаление всех лимитов карты» добавлено описание: - «Метод удаляет с карты все лимиты (количественные ограничители) включая ограничители по времени»
22.03.2023	Добавлены методы 2.4.10.2 «Удаление ограничителя по времени для карты», 2.4.10.3 «Создание ограничителя по времени для карты» и 2.4.10.4 «Изменение ограничителя по времени для карты» - Указанные методы позволяют управлять ограничителями по времени для карты
22.03.2023	Добавлен метод 2.3.3 «Получение списка ограничителей по времени для договора»
15.05.2023	Метод 2.4.8 изменил номер на 2.4.8.1 Добавлены методы по работе с виртуальными картами: 2.4.8.2 – Создание виртуальной карты 2.4.8.3 – Получение информации о заявках на создание виртуальных карт 2.4.8.4 – Получение информации о виртуальной карте 2.4.8.5 – Изменение Ф.И.О. и/или телефона виртуальной карты
03.07.2023	Добавлена информация о группе карт в методы 2.4.1 «Список карт по договору», 2.5.1 «Получение списка операций по договору за период» и 2.5.2 «Получение списка операций по договору по дате изменения»
13.07.2023	HTTP-заголовок RnCard-RequestId – уникальный идентификатор запроса.
03.08.2023	В описание метода 2.4.8.2 «Создание виртуальной карты» добавлено описание к параметру EDate.
30.08.2023	В возвращаемый набор данных метода 2.4.2.1 - «Получение списка лимитов карты» добавлены следующие поля: MaxNumber - Максимальное количество транзакций. 0 – Не установлено. CurNumber - Доступное количество транзакций. В случае, если значение поля MaxNumber равно нулю, то значение поля CurNumber не имеет смысла.
30.08.2023	Добавлено поле FlagRead в возвращаемый набор данных в следующих методах: 2.3.3 - «Получение списка ограничителей по времени для договора» 2.4.2.1 – «Получение списка лимитов карты» 2.4.3.1 – «Получение списка товарных ограничителей карты» 2.4.10.1 – «Получение списка ограничителей по времени для карты»  Возможные значения поля FlagRead: Y – Изменение лимита/ограничителя запрещено. N – Изменение лимита/ограничителя разрешено.
30.08.2023	Следующие методы перешли на версию 2: 2.4.2.1 – «Получение списка лимитов карты» 2.6.8 – «Получение информации о лимитах и/или ограничителях всех карт договора, одной карты и/или договора»
20.12.2023	Убрано описание небезопасного способа передачи пароля

## СОДЕРЖАНИЕ

<b>1</b>	<b>ВВОДНАЯ ЧАСТЬ .....</b>	<b>6</b>
1.1	ПРЕДНАЗНАЧЕНИЕ ДОКУМЕНТА .....	6
1.2	ПРИНЯТЫЕ СОКРАЩЕНИЯ.....	6
1.3	ПЕРЕДАЧА ПАРОЛЯ .....	6
1.4	БЕЛЫЙ СПИСОК IP-АДРЕСОВ .....	6
1.5	РЕКОМЕНДАЦИИ .....	6
1.6	Код договора и аутентификация.....	7
1.7	Дополнительные заголовки запросов .....	7
<b>2</b>	<b>ИНТЕГРАЦИОННЫЕ ИНТЕРФЕСЫ .....</b>	<b>8</b>
2.1	АДРЕС В СЕТИ ИНТЕРНЕТ .....	8
2.2	ОПИСАНИЕ ОШИБОК.....	8
2.3	ДОГОВОР.....	9
2.3.1	<i>Баланс договора (информация по договору).....</i>	<i>9</i>
2.3.2	<i>Региональные ограничители договора.....</i>	<i>10</i>
2.3.2.1	Получение списка региональных ограничителей договора.....	13
2.3.2.2	Создание/Изменение/Удаление регионального ограничителя договора .....	14
2.3.3	<i>Получение списка ограничителей по времени для договора .....</i>	<i>16</i>
2.4	КАРТЫ .....	18
2.4.1	<i>Список карт по договору.....</i>	<i>18</i>
2.4.2	<i>Лимиты (количественные ограничители) .....</i>	<i>19</i>
2.4.2.1	Получение списка лимитов карты .....	19
2.4.2.2	Создание лимита .....	20
2.4.2.3	Изменение лимита.....	21
2.4.2.4	Удаление лимита .....	22
2.4.2.5	Удаление всех лимитов карты .....	22
2.4.3	<i>Товарные ограничители.....</i>	<i>23</i>
2.4.3.1	Получение списка товарных ограничителей карты .....	23
2.4.3.2	Создание товарного ограничителя карты .....	25
2.4.3.3	Удаление товарного ограничителя.....	26
2.4.3.4	Удаление всех товарных ограничителей карты .....	27
2.4.3.5	Изменение товарного ограничителя.....	27
2.4.4	<i>Блокировка карт.....</i>	<i>29</i>
2.4.5	<i>Разблокировка карт.....</i>	<i>29</i>
2.4.6	<i>Получение информации о статусе карт .....</i>	<i>30</i>
2.4.7	<i>Количество попыток ввода пин-кода карты .....</i>	<i>31</i>
2.4.7.1	Получение информации о количестве попыток ввода пин-кода .....	31
2.4.7.2	Сброс количества попыток ввода пин-кода карты .....	32
2.4.8	<i>Виртуальная карта .....</i>	<i>33</i>
2.4.8.1	Закрытие виртуальной карты.....	33
2.4.8.2	Создание виртуальной карты .....	34
2.4.8.3	Получение информации о заявках на создание виртуальных карт .....	35
2.4.8.4	Получение информации о виртуальной карте.....	37
2.4.8.5	Изменение Ф.И.О. и/или телефона у виртуальной карты.....	37
2.4.9	<i>Региональные ограничители карты .....</i>	<i>38</i>
2.4.9.1	Получение списка региональных ограничителей карты .....	38
2.4.9.2	Создание/Изменение/Удаление регионального ограничителя карты.....	39
2.4.9.3	Удаление всех региональных ограничителей карты .....	41
2.4.10	<i>Ограничители по времени для карты.....</i>	<i>42</i>
2.4.10.1	Получение списка ограничителей по времени для карты .....	42
2.4.10.2	Удаление ограничителя по времени для карты .....	43
2.4.10.3	Создание ограничителя по времени для карты.....	44
2.4.10.4	Изменение ограничителя по времени для карты .....	45
2.5	ОПЕРАЦИИ .....	47
2.5.1	<i>Получение списка операций по договору за период .....</i>	<i>47</i>
2.5.2	<i>Получение списка операций по договору по дате изменения .....</i>	<i>49</i>
2.6	ПРОЧИЕ.....	49
2.6.1	<i>Получение списка услуг .....</i>	<i>49</i>
2.6.2	<i>Получение статуса заявки .....</i>	<i>50</i>
2.6.3	<i>Получение кода договора по номеру карты.....</i>	<i>52</i>
2.6.4	<i>Получение информации о странах и их регионах .....</i>	<i>53</i>

	ТР _____	
--	----------	--

2.6.5	Получение списка партнёров .....	54
2.6.6	Получение списка НПО .....	55
2.6.7	Получение списка ТО .....	56
2.6.8	Получение информации о лимитах и/или ограничителях всех карт договора, одной карты и/или договора .....	59
2.7	ГРУППЫ КАРТ .....	62
2.7.1	Получение списка групп карт .....	62
2.7.2	Управление группами карт .....	63
2.7.3	Управление картами в группах .....	64

## СПИСОК ТАБЛИЦ

Таблица 1.	Коды ошибок.....	9
Таблица 2.	Формат ответа в случае ошибки.....	9
Таблица 3.	Входные параметры метода, возвращающего информацию по договору .....	9
Таблица 4.	Структура ответа метода, возвращающего информацию по договору .....	10
Таблица 5.	Описание входящих параметров запроса списка карт по договору.....	18
Таблица 6.	Структура ответа на запрос списка карт по договору .....	18
Таблица 7.	Описание входящих параметров запроса для получения списка лимитов .....	19
Таблица 8.	Описание структуры ответа на запрос получения списка лимитов карты .....	20
Таблица 9.	Описание входящих параметров запроса на получение списка лимитов .....	20
Таблица 10.	Описание входящих параметров запроса на изменение лимита .....	21
Таблица 11.	Описание входящих параметров запроса на удаление лимита .....	22
Таблица 12.	Описание входящих параметров запроса на товарные ограничители карты .....	23
Таблица 13.	Описание результата запроса товарных ограничителей карты.....	24
Таблица 14.	Описание входящих параметров запроса создания товарного ограничителя.....	25
Таблица 15.	Описание входящих параметров запроса удаления товарного ограничителя.....	26
Таблица 16.	Описание входящих параметров метода блокировки карты .....	29
Таблица 17.	Описание входящих параметров метода разблокировки карты .....	30
Таблица 18.	Описание входящих параметров метода управления группами карт .....	30
Таблица 19.	Описание входящих параметров метода блокировки карты .....	31
Таблица 20.	Описание входящих параметров метода блокировки карты .....	32
Таблица 21.	Описание входящих параметров метода блокировки карты .....	33
Таблица 22.	Описание входящих параметров запроса операций по договору за период .....	47
Таблица 23.	Описание результата выполнения запроса карт договора .....	48
Таблица 24.	Описание входящих параметров запроса операций по договору по дате изменения .....	49
Таблица 25.	Описание входящих параметров метода получения списка услуг.....	49
Таблица 26.	Описание входящих параметров метода получения списка услуг.....	50
Таблица 27.	Описание входящих параметров метода получения статуса заявки .....	50
Таблица 28.	Описание результата метода получения статуса заявки .....	51
Таблица 29.	Описание входящих параметров метода получения кода договора по его карте .....	52
Таблица 30.	Описание результата метода получения кода договора по его карте.....	52
Таблица 31.	Описание результата метода получения кода договора по его карте.....	54
Таблица 32.	Описание входящих параметров метода получения списка групп карт .....	62
Таблица 33.	Описание полей возвращаемых методом получения списка групп карт .....	63
Таблица 34.	Описание входящих параметров метода управления группами карт .....	63

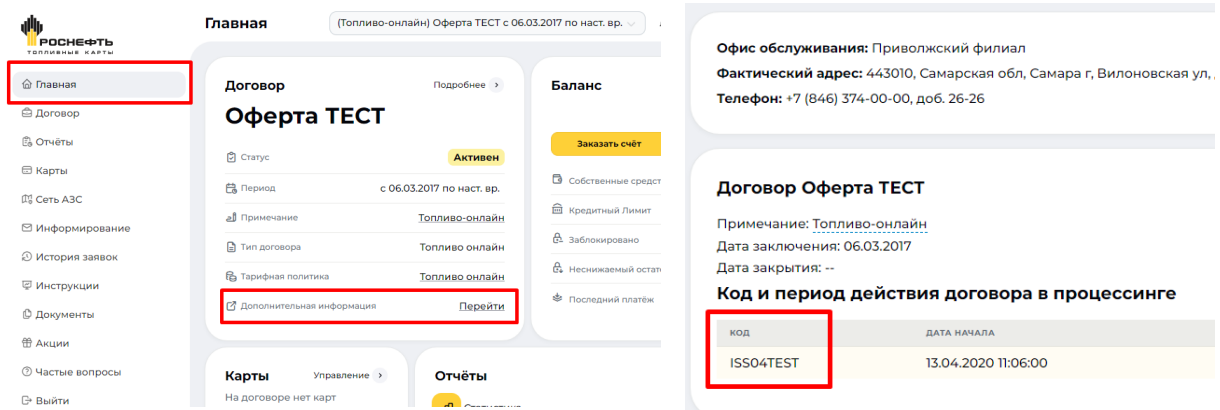
# 1 ВВОДНАЯ ЧАСТЬ

## 1.1 Предназначение документа

Документ описывает интерфейсы взаимодействия/интеграции сторонних систем с информационными системами РН-Карт. Взаимодействие осуществляется по технологии WebAPI (HTTP GET/POST запросы на веб-узел компании и получение ответов в формате XML либо JSON).

Пароль и логин для веб-сервиса необходимо взять от учетной записи личного кабинета.

Чтобы получить код договора – нужно вызвать метод: 2.6.3 *GetContractByCard*. Или в ЛК в разделе «Дополнительная информация».



## 1.2 Принятые сокращения

БД	База данных
ПО	Программное обеспечение
ЛК	Личный кабинет
Web API	Интерфейс программирования приложений, состоящий из определенного набора HTTP-запросов и определённой структуры HTTP-ответов, для выражения которых используют XML или JSON форматы

## 1.3 Передача пароля

Для передачи пароля в заголовке HTTP-запроса, необходимо отправить заголовок следующего вида:

Поле заголовка(ключ)	Значение
RnCard-Identity-Account-Pass	Пароль или md5-хеш пароля, закодированные в Base64

Пример PowerShell

```
$result1 = $null
$uri = 'https://lkapi.rn-card.ru/api/emv/v2/GetOperByContract?u=ЛОГИН&contract=ISS00000&begin=2020-02-01T00:00:00&end=2020-02-29T23:59:59'
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$headers = @{
    'RnCard-Identity-Account-Pass' = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes('ПАРОЛЬ'))
}
$result1 = Invoke-RestMethod -Uri $uri -Method Get -Headers $headers
$result1
```

## 1.4 Белый список IP-адресов

За учётной записью в ЛК, закрепляются IP-адреса, с которых разрешены запросы для указанной учётной записи.

Для использования WebAPI, клиент должен, на официальном бланке организации, сообщить список IP-адресов, с которых будут осуществляться запросы с привязкой к учётной записи ЛК.

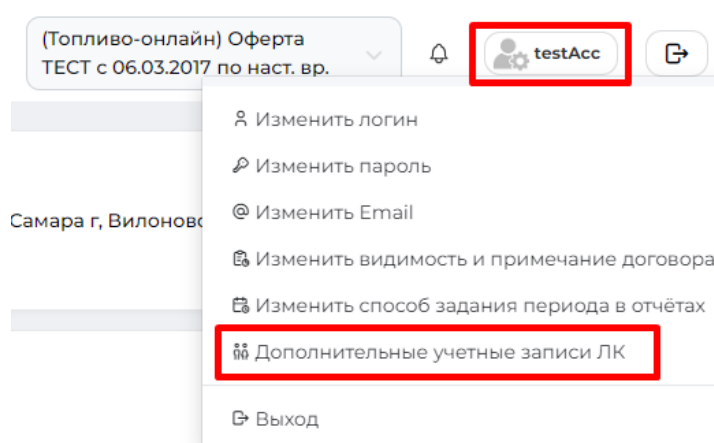
Белый список IP – опция, если он не задан, то разрешены запросы с любого IP-адреса.

## 1.5 Рекомендации

- Использовать для доступа к WebAPI учётную запись ЛК отличную от той, что используется для входа в Веб-интерфейс ЛК, потому что неправильный ввод пароля на странице входа ЛК приведёт к следующим событиям:

- на форме ввода будет запрошен ввод CAPCHA
- доступ по WebAPI будет заблокирован, ДО ввода правильного пароля на форме входа в ЛК.

Создать дополнительную учетную запись для договора можно в ЛК в разделе «Дополнительные учетные записи ЛК».



- 2) Использовать WebAPI разумно! Бесконечный опрос сервиса будет воспринят как DDOS-атака и доступ будет **заблокирован**! Лучше уточнить в поддержке, каким образом оптимально решить ту или иную задачу.
- 3) Все национальные символы, передаваемые в параметрах GET запроса, следует подвергать кодированию при помощи функции urlencode

## 1.6 Код договора и аутентификация

Логин и пароль для веб-сервиса необходимо взять от учётной записи личного кабинета.  
Код договора можно посмотреть в ЛК (см. п 1.1)

## 1.7 Дополнительные заголовки запросов

Опционально, http-запросы к WebAPI можно добавлять следующие заголовки:

http-заголовок	Значение	Примечание
RnCard-RequestId	Уникальный идентификатор запроса.	Клиент на своей стороне генерирует уникальный идентификатор запроса, например, GUID. Этот идентификатор поможет найти запрос в журнале, при разборе технических моментов ИТ-службой при обращении клиента.

## 2 ИНТЕГРАЦИОННЫЕ ИНТЕРФЕСЫ

### 2.1 Адрес в сети Интернет

Адрес API личного кабинет клиента (<https://lkapi.rn-card.ru>).

Протокол HTTP не поддерживается.

Поддерживаются TLS1.2.

SSLv3, TLS1.0 и TLS1.1 не поддерживаются.

### 2.2 Описание ошибок

В случае ошибки формируется сообщение вида - [КОД] ТЕКСТ [RequestId=ID]

Текст ошибки может содержать идентификатор заявки, если заявка была создана в ЛК, но по какой-то причине не удалось её выполнить в процессинге, например, “[6] Bad request [RequestId=201701]”

Сообщение об ошибке	Описание	HTTP код
[0] Success	Запрос выполнен успешно	404 Not Found
[1] The user is not found	Пользователь не найден	404 Not Found
[2] The contract is not found	Договор не найден	404 Not Found
[3] The card is not found	Карта не найдена	404 Not Found
[4] The card is block	Карта заблокирована	403 Forbidden
[5] The card is not tied to a contract	Карта не привязана к указанному договору	403 Forbidden
[6] Bad request	Неизвестная ошибка	403 Forbidden
[7] Period is bad	Неверный период (неверный формат)	400 BadRequest
[8] User or password is bad	Неверный пароль	403 Forbidden
[9] Period is big	Неверный период (больше 2х месяцев)	400 BadRequest
[10] Card already is block	Карта уже заблокирована	400 BadRequest
[11] Card already is unblock	Карта уже разблокирована	400 BadRequest
[12] Request status error	Ошибка при получении статуса заявки	400 BadRequest
[13] Request not registered	Заявка не зарегистрирована	400 BadRequest
[14] Access forbidden	Доступ запрещён	400 BadRequest
[15] Incorrect phone number format	Неверный формат телефона	400 BadRequest
[16] Incorrect date	Неверный формат даты	400 BadRequest
[17] Incorrect cardholder name	Некорректно указан владелец карты/водитель	400 BadRequest
[18] Incorrect value	Некорректная величина операции	400 BadRequest
[19] Limit code not found	Не указан код лимита	400 BadRequest
[20] Contract service not payed	Не оплачена услуга «Информационное обслуживание»	403 Forbidden
[21] Card group not found	Группы карт не найдены	204 NoContent
[22] Incorrect group name	Название группы не указано или указано некорректно	400 BadRequest
[23] Incorrect group id	ID группы не указан или указан некорректно	400 BadRequest
[24] Card group contains cards	Группа карт содержит карты	400 BadRequest
[25] Unknown action	Неизвестная операция	400 BadRequest
[26] The cards do not belong to the current contract or are not in the specified group	Карты не принадлежат договору или не находятся в указанной группе	400 BadRequest
[27] Not specified card group to transfer	Не указана группа для перемещения карт	400 BadRequest
[28] The cards do not belong to the current contract or are already in the specified group	Карты не принадлежат договору или уже находятся в указанной группе	400 BadRequest
[29] The card is not virtual	Карта не является виртуальной	400 BadRequest
[30] Virtual card info is not defined	Не указана информация о виртуальной карте	400 BadRequest
[31] Mobile phone is not valid	Некорректный номер мобильного телефона	400 BadRequest
[32] There is nothing to change. The current and requested data is equal	Запрашиваемое изменение не повлияет на исходные данные. Т.к. результат изменения идентичен текущим данным.	400 BadRequest



	TP _____	
--	----------	--

[33] WebApi have been banned for current contract.	Временно запрещён вызов методов WebApi для договора. Для уточнения причины просьба связаться с тех. поддержкой или менеджером.	403 Forbidden
[34] User account was locked	Учётная запись заблокирована	403 Forbidden
[35] User account was locked. Change password to unlock	Учётная запись заблокирована из-за слабого пароля. После смены пароля учётная запись будет разблокирована.	403 Forbidden
[36] Access forbidden. Your Ip-address not in white-list	Вызов методов WebAPI недоступен пользователю. IP-адрес, с которого происходит вызов метода не находится в белом списке.	403 Forbidden
[37] Request with virtual card is not permitted by contract	Работа с виртуальными картами запрещена для договора.	403 Forbidden

Таблица 1. Коды ошибок

Формат ответа в случае ошибки:

Формат результата	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;Error&gt;   &lt;Message&gt;[2] The contract is not found&lt;/Message&gt; &lt;/Error&gt;</pre>
JSON	<pre>{"Message": "[2] The contract is not found"}</pre>

Таблица 2. Формат ответа в случае ошибки

## 2.3 Договор

### 2.3.1 Баланс договора (информация по договору)

Метод, возвращающий данные о балансе договора в формате XML (по умолчанию) или JSON.

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetContractBalance*

Адрес метода: */api/emv/v1/GetContractBalance?u={u}&contract={contract}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора (можно получить методом 2.6.3.)
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json;

Таблица 3. Входные параметры метода, возвращающего информацию по договору

#### Возвращаемый результат

Формат результата	Результат
XML	<pre>&lt;Contract Balance="9304326.68" Available="9303327.68" CreditLimit="0" Blocked="999" AddLimit="0"/&gt;</pre>
JSON	<pre>{   "Balance": 9304326.68,   "Available": 9303327.68,   "CreditLimit": 0.0,   "Blocked": 999.0,   "AddLimit": 0.0 }</pre>
Значения полей	
Balance	Собственные средства
Available	Доступно средств
CreditLimit	Кредитный лимит
Blocked	Заблокировано

	ТР _____	
--	----------	--

AddLimit	Неснижаемый остаток
----------	---------------------

Таблица 4. Структура ответа метода, возвращающего информацию по договору

## 2.3.2 Региональные ограничители договора

**Внимание.** Во всех примерах, если не указано обратное, запрещены партнёры без эквайринговых данных из-за их особенности работать поверх остальных региональных ограничителей.

### Правила работы региональных ограничителей

1. По умолчанию, при отсутствии региональных ограничителей, система разрешает обслуживание на всех ТО.

2. При установке разрешающих ограничителей на определенные АЗС / НПО / Регионы / Страну / Партнёра автоматически будут ЗАПРЕЩЕНЫ все остальные точки обслуживания, на которых не установлен разрешающий ограничитель.

**Исключение:** разрешающие ограничители договора по партнёрам без эквайринговых данных (Type = «Н» в методе GetPartners [п. 2.6.5](#)) не запрещают обслуживание на точках обслуживания, на которых не установлен разрешающий ограничитель.

Например, на карте установлен только один региональный ограничитель, разрешающий обслуживание в Москве. Обслуживание будет разрешено только в Москве, все остальные точки обслуживания России и других стран будут автоматически запрещены.

3. Запрещающие ограничители, в отличие от разрешающих, не меняют поведение системы при обслуживании на остальных ТО.

Пример 1. На карте установлен только один региональный ограничитель, запрещающий обслуживание в Москве. Обслуживание будет запрещено только в Москве, на остальных точках обслуживания поведение системы не изменится и обслуживание будет возможно согласно пункту 1.

Пример 2. Первый ограничитель разрешает обслуживание на АЗС1, второй ограничитель запрещает обслуживание на АЗС2. Обслуживание будет разрешено только на АЗС1. Остальные ТО будут автоматически запрещены по пункту 2, как не попадающие в разрешающие ограничители.

4. Возможна установка нескольких ограничителей карты или договора. Обслуживание будет запрещено, если хоть один из ограничителей запрещает обслуживание.

Пример 1: Если первый ограничитель разрешает обслуживание в Москве, а второй ограничитель – запрещает обслуживание на ТО №123 Москвы, то будет разрешено обслуживание только на ТО Москвы, кроме ТО №123.

Пример 2: Если первый ограничитель запрещает обслуживание в Москве, а второй ограничитель – разрешает обслуживание на ТО №123 Москвы, то обслуживание нигде не будет разрешено.

Пример 3: Если первый ограничитель запрещает обслуживание в Москве, а второй ограничитель – разрешает обслуживание на ТО №234 Краснодарского края, то обслуживание по карте будет разрешено только на ТО №234.

Пример 4: Если первый ограничитель разрешает «Россия», второй – «Московская область», третий – «НПО Рязаньнефтепродукт», то обслуживание возможно на всех ТО России.

5. Для проверки возможности обслужиться на ТО, процессинг проверяет все ограничители и карты, и договора. Обслуживание будет разрешено, если все ограничители карты и договора не запретят обслуживание.

Например, региональные ограничители карты разрешают обслуживание только на ТО партнёра, а региональные ограничители договора разрешает обслуживание только на ТО Роснефть. Обслуживание по этой карте нигде не будет разрешено.

6. В одном ограничителе можно указывать несколько параметров. В этом случае запрет/разрешение обслуживание будет высчитываться по логическому «И» всех параметров ограничителя.

	2.0.0.32	10 / 65
--	----------	---------

	ТР _____	
--	----------	--

Например, У НПО000024 (Рязаньнефтепродукт) есть терминалы в 3х регионах: Владимирская, Московская и Рязанская область. Если создать разрешающий ограничитель «Россия / Московская область / НПО Рязаньнефтепродукт», то обслужиться получится только на ТО московской области, которые относятся к НПО Рязаньнефтепродукт.

7. Рекомендуется использовать перечисления параметров в одном ограничителе, вместо создания нескольких ограничителей по каждому параметру. Управление ограничителями – ресурсоёмкие операции и время выполнения процедуры создания ограничителя по 1й АЗС такое же, как и на создания ограничителя по 200 АЗС.

Например, необходимо разрешить обслуживание только в 20 регионах России. Рекомендуется создать один ограничитель с перечислением 20 регионов, чем 20 ограничителей по каждому региону.

### Партнёры

Получить список партнёров можно методом GetPartners [п. 2.6.5](#).

Процессинг работает с партнёрами 2х типов:

1. Партнёры, с которыми построена работа через эквайринговые данные.

В процессинг РН-Карт загружена информация по всем ТО партнёра. Есть возможность устанавливать ограничители по странам, регионам, НПО и ТО партнёра.

Внимание. РН-Карт не проверяет предоставляемую партнёром информацию по ТО, не контролирует ежедневное автоматическое изменение информации. Партнёр по своему желанию может изменить, удалить, пересоздать ТО, НПО. Если партнёр пересоздаст ТО, то все ограничители по этой ТО будут недействительны. РН-Карт не рекомендует создавать региональные ограничители по ТО, НПО партнёра и не рекомендует создавать ограничители по странам и регионам ТО партнёра.

Например, партнёр при перезаключении договора с ТО, пересоздаёт в своём процессинге ТО с другим кодом. В процессинге РН-Карт удалится старая ТО и создастся новая. Все запрещающие региональные ограничители по старой ТО будут недействительны и не будет запрещено обслуживание на новых ТО.

2. Партнёры, с которыми построена работа без эквайринговых данных (межхостовые партнёры).

В процессинг РН-Карт не загружена информация по ТО партнёра.

Ограничения:

1. Можно запретить партнёра только целиком, будут запрещены все ТО партнёра.
2. Можно разрешить партнёра целиком, будут разрешены все ТО партнёра.
3. Если на договоре нет ограничителя по партнёру без эквайринговых данных, то по умолчанию партнёр разрешен.
4. Такого партнёра нельзя указать для ограничителя карты, только для ограничителя договора.

Ограничители по партнёру без эквайринговых данных работают поверх остальных региональных ограничителей, поэтому попадают во все вышеуказанные исключения.

Пример 1. Если на договоре установлен только один разрешающий ограничитель «Московская область», то будут доступно обслуживание на всех ТО московской области и на ТО всех партнёров без эквайринговых данных по всей России и в других странах.

Пример 2. Если на договоре установлен только разрешающий ограничитель по партнёру без эквайринговых данных «Партнёр1», то обслуживание разрешено на всех ТО, в том числе на всех ТО партнёров.

Пример 3. Если на договоре установлен только разрешающий ограничитель по партнёру с эквайринговыми данными «Партнёр2», то обслуживание разрешено на ТО «Партнёра 2» и на всех ТО партнёров, которые работают без эквайринговых данных.

Внимание. РН-Карт рекомендует периодически проверять списки партнёров на появление новых партнёров и изменения типов партнёров и сверять с установленными у клиентов региональными ограничителями. Возможны ситуации, когда клиенту станут доступны ТО и партнёры, которые ранее были запрещены.

Пример 1. На договоре разрешено обслуживание только на АЗС Роснефть на территории Московской области. Для этого установлены следующие региональные ограничители:

1. Разрешающий ограничитель «Россия / Московская область».

	2.0.0.32	11 / 65
--	----------	---------

	ТР _____	
--	----------	--

2. Разрешающие ограничители со списком НПО Роснефти.
3. Запрещающие ограничители по всем партнёрам, которые работают без эквайринговых данных.

Если в процессинге подключить нового партнёра без эквайринговых данных и не добавить на договор запрещающий ограничитель по новому партнёру, то будет разрешено обслуживание на ТО Роснефти на территории Московской области и на всех ТО нового партнёра в любом регионе России и в любой стране.

Если в процессинге подключить нового партнёра с эквайринговыми данными и не добавить на договор запрещающий ограничитель, то новый партнёр будет автоматически запрещен и будет разрешено обслуживание только на ТО Роснефти на территории Московской области.

Пример 2. На договоре разрешено обслуживание только на «Партнёре 1» - партнёре без эквайринговых данных. Для этого установлены следующие региональные ограничители:

1. Запрещающие ограничители по всем партнёрам, которые работают без эквайринговых данных, кроме «Партнёра 1»
2. Или разрешающий ограничитель по «Партнёру 1», или отсутствие ограничителя по «Партнёру 1»
3. Запрещающий ограничитель со списком всех стран.

При изменении партнёра на работу через эквайринговые данные фактически будет создан новый партнёр, а старый станет недействительным. И если на клиентский договор не добавить разрешающий ограничитель по «новому» «Партнёру 1», то обслуживание по нему не будет разрешено.

### Страны, регионы

Получить список стран с их регионами можно методом GetCountryList [п. 2.6.4.](#)

В одном региональном ограничителе можно указывать регионы разных стран. При установке регионального ограничителя по регионам стран, обязательно указывать страны, к которым эти регионы относятся. Если указать страну, но не указать ни один регион этой страны, то автоматически учитываться все регионы страны.

Пример 1. Чтобы разрешить обслуживание только на территории Московской и Рязанской областей, можно или добавить один разрешающий ограничитель «Россия / Московская область, Рязанская область», или добавить 2 разрешающих ограничителя «Россия / Московская область» и «Россия / Рязанская область». Оба варианта равносильны.

Пример 2. Разрешить обслуживание только на территории Московской и Рязанской областей России и Минской области республики Беларусь можно несколькими способами:

1. Добавить ограничитель «Россия, Беларусь / Московская область, Рязанская область, Гомельская область»
2. Добавить 2 ограничителя: «Россия / Московская область, Рязанская область» и «Беларусь / Гомельская область»
3. Добавить 3 ограничителя: «Россия / Московская область», «Россия / Рязанская область» и «Беларусь / Гомельская область»

Пример 3. Запретить обслуживание на всей территории республики Беларусь и Смоленской области РФ можно двумя способами:

1. Добавить ограничитель «Россия, Беларусь / Смоленская область, Минская область, Брестская область, Витебская область итд»
2. Добавить 2 ограничителя: «Россия / Смоленская область» и «Беларусь»

### НПО, ТО

Получить список НПО можно методом GetNPOList [п. 2.6.6.](#), список ТО - GetPOSList [п. 2.6.7](#)

НПО – группировка ТО по поставщику, предприятию нефтепродуктообеспечения или другому признаку. Например «Томскнефтепродукт», «Башнефть», «Чужие точки обслуживания», «Частные точки обслуживания РН-Запад».

Партнёры, работающие через эквайринговые данные, также ведут списки НПО в процессинге РН-Карт. Но, как описывалось выше, РН-Карт не рекомендует устанавливать ограничители по НПО и ТО партнёров.

Чтобы установить ограничитель по всем ТО Роснефти, нужно перечислить все НПО Роснефти.

**Внимание.** РН-Карт рекомендует периодически актуализировать списки НПО в своих системах для недопущения установки некорректных региональных ограничителей.

	2.0.0.32	12 / 65
--	----------	---------

	ТР _____	
--	----------	--

Ограничение: Максимальное количество НПО в одном ограничителе – 10.

Если нужно запретить/разрешить больше 10 НПО (например, все НПО Роснефти), то рекомендуется разбить список НПО на группы по 10 НПО + группа с остатком и создать ограничитель для каждой группы НПО.

### 2.3.2.1 Получение списка региональных ограничителей договора

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetContractRegionRestriction*

Адрес метода: */api/emv/v2/GetContractRegionRestriction?u={u}&contract={contract}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RestrictionList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;Item CountryCode="RUS" RegionCode="01,11" NpoCode="" PosCode="" RestrictionTypeCode="E" RestrictionID="259483" FlagRead="N" /&gt;   &lt;Item CountryCode="" RegionCode="" NpoCode="" PosCode="AZS101309" RestrictionTypeCode="D" RestrictionID="261103" FlagRead="N" PartnerCode="" /&gt; &lt;/RestrictionList&gt;</pre>
JSON	<pre>{   "RestrictionList": [     {       "CountryCode": "RUS",       "RegionCode": "01,11",       "NpoCode": "",       "PosCode": "",       "RestrictionTypeCode": "E",       "RestrictionID": "259483",       "FlagRead": "N",       "PartnerCode": ""     },     {       "CountryCode": "",       "RegionCode": "",       "NpoCode": "",       "PosCode": "AZS101309",       "RestrictionTypeCode": "D",       "RestrictionID": "261103",       "FlagRead": "N",       "PartnerCode": ""     }   ] }</pre>
Описание полей	
CountryCode	Трёхбуквенный код страны. Может содержать несколько кодов указанных через разделитель «,»(запятая).

	2.0.0.32	13 / 65
--	----------	---------

	ТР _____	
--	----------	--

RegionCode	Код региона Может содержать несколько кодов указанных через разделитель «,»(запятая).
NpoCode	Код НПО. Может содержать несколько кодов указанных через разделитель «,»(запятая).
PosCode	Код ТО Может содержать несколько кодов указанных через разделитель «,»(запятая).
RestrictionType	Тип ограничителя. Возможные значения: E – Обслуживание разрешено, D – Обслуживание запрещено.
RestrictionID	ID ограничителя
FlagRead	Флаг: Y – Изменение ограничителя запрещено. N – Изменение ограничителя разрешено.
PartnerCode	Код партнёра

Описание структуры ответа на запрос получения списка региональных ограничителей карты

### 2.3.2.2 Создание/Изменение/Удаление регионального ограничителя договора

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: ContractRegionRestriction

Адрес метода: /api/emv/v2/ContractRegionRestriction/

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Act	Да	Действие с региональным ограничителем: C – Создать, M – Изменить, D – Удалить.
RestrictionType	Усл. обяз	Тип ограничителя. Возможные значения: E – Обслуживание разрешено, D – Обслуживание запрещено. Параметр является обязательным при создании или изменении регионального ограничителя.
CountryCode	Нет	Трёхбуквенный код страны. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 60.
RegionCode	Нет	Код региона. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 30. В параметре CountryCode необходимо указать страны, регионы которых перечислены в RegionCode.
PartnerCode	Нет	Код партнёра. Если указан межхостовой партнёр (без эквайринговых данных, тип партнёра «Н»), то параметры CountryCode, RegionCode, NpoCode и PosCode игнорируются.
PosCode	Нет	Код ТО. Может содержать несколько кодов ТО(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 200.
NpoCode	Нет	Код НПО. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 10.

	2.0.0.32	14 / 65
--	----------	---------

	ТР _____	
--	----------	--

RestrictionCode	Усл. обяз	Код удаляемого или изменяемого ограничителя. Параметр является обязательным только при удалении или изменении регионального ограничителя.
-----------------	-----------	--

*Описание входящих параметров запроса на создание регионального ограничителя*

Пример ограничителя:

Act = C

RegionCode = 11 (Архангельская область)

CountryCode = RUS

RestrictionType = D

С указанными параметрами будет создан региональный ограничитель, по которому будет запрещено обслуживание в Архангельской области.

В случае если код региона не будет принадлежать ни одной стране, то будет выдана ошибка.

Заполнять поля RestrictionType, CountryCode, RegionCode, PartnerCode, PosCode, NpoCode при удалении ограничителя необходимо для сохранения старых значений ограничителя в журнале изменения регионального или межхостового ограничителей.

При изменении ограничителя, ограничитель будет изменён в соответствии со всеми новыми значениями параметров. Если параметр оставить пустым, то у ограничителя параметр станет пустым. Например, если при изменении ограничителя «Россия/Новосибирская область» параметр RegionCode оставить пустым, то ограничитель будет изменён на «Россия».

Ограничения:

1. Нельзя изменять ограничители не по межхостовым партнёрам (без эквайринговых данных) на ограничители по межхостовым партнёрам.
2. Для ограничителей по межхостовым партнёрам (без эквайринговых данных) возможно изменение только типа ограничителя (запрещено/разрешено обслуживание).

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RegionRestrictionResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"&gt;   &lt;RequestDetail&gt;     &lt;RequestResponse&gt;       &lt;RequestID&gt;52450556&lt;/RequestID&gt;       &lt;ErrorCode&gt;0&lt;/ErrorCode&gt;       &lt;ErrorMessage&gt;[0] Запрос выполнен успешно&lt;/ErrorMessage&gt;       &lt;LimitCode&gt;67465802&lt;/LimitCode&gt;     &lt;/RequestResponse&gt;   &lt;/RequestDetail&gt; &lt;/RegionRestrictionResponse&gt;</pre>
JSON	<pre>{   "RegionRestrictionResponse": {     "RequestDetail": [{       "RequestResponse": {         "RequestID": 52450556.0,         "ErrorCode": 0,         "ErrorMessage": "[0] Запрос выполнен успешно",         "LimitCode": "67465802"       }     }]   } }</pre>
Описание полей	
Contract	Код договора
Card	Графический номер карты
RequestID	ID заявки
ErrorCode	Код ошибки

	2.0.0.32	15 / 65
--	----------	---------

	ТР _____	
--	----------	--

ErrorMessage	Сообщение об ошибке
RestrictionID	ID ограничителя

Описание структуры ответа на запрос создания регионального ограничителя карты

### 2.3.3 Получение списка ограничителей по времени для договора

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetContractTimeRestriction*

Адрес метода: */api/emv/v2/GetContractTimeRestriction?u={u}&contract={contract}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;TimeRestrictions xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"&gt;   &lt;Item Code="TEMPL_RLI_CURR9" WeekDays="2" TimeInterval="00002359" FlagRead="N" /&gt;   &lt;Item Code="TEMPL_RLI_CURR11" WeekDays="3" TimeInterval="00001204" FlagRead="N" /&gt;   &lt;Item Code="TEMPL_RLI_CURR12" WeekDays="3" TimeInterval="15062359" FlagRead="N" /&gt; &lt;/TimeRestrictions&gt;</pre>
JSON	<pre>{   "TimeRestrictions": [     {       "Code": "TEMPL_RLI_CURR9",       "WeekDays": "2",       "TimeInterval": "00002359",       "FlagRead": "N"     },     {       "Code": "TEMPL_RLI_CURR11",       "WeekDays": "3",       "TimeInterval": "00001204",       "FlagRead": "N"     },     {       "Code": "TEMPL_RLI_CURR12",       "WeekDays": "3",       "TimeInterval": "15062359",       "FlagRead": "N"     }   ] }</pre>
Описание полей	
Code	Код лимита (уникальный идентификатор в пределах карты)
WeekDays	День недели работы лимита (1 – Пн., 2 – Вт., 3- Ср., 4 – Чт., 5 – Пт., 6 – Сб., 7 – Вс.)
TimeInterval	Временной интервал которым ограничено обслуживание. Указывается в часах и минутах без разделителя. Например, интервал «15062359»

	2.0.0.32	16 / 65
--	----------	---------



	ТР _____	
--	----------	--

	ограничивает обслуживание с 15 часов 06 минут до 23 часов 59 минут включительно.
FlagRead	Флаг: Y – Изменение ограничителя запрещено. N – Изменение ограничителя разрешено.

*Описание структуры ответа на запрос получения списка ограничителей по времени для договора*

2.4 Карты

2.4.1 Список карт по договору

Метод возвращает список карт договора.

Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetCardsByContract*

Адрес метод: */api/emv/v1/GetCardsByContract?u={login}& contract={contract}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json;

Таблица 5. Описание входящих параметров запроса списка карт по договору

Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;ArrayOfCard&gt;   &lt;Card Num="0000000000000007" Rem="Prokofiev Vadim" SName="Заблокирована (доставка клиенту)" SCode="05_D" CardGrp="Test" /&gt;   &lt;Card Num="0000000000000006" Rem="Grigoryev Denis" SName="В работе" SCode="00" CardGrp="Test" /&gt; &lt;/ArrayOfCard&gt;</pre>
JSON	<pre>[   {     "Num": "0000000000000007",     "Rem": "Prokofiev Vadim",     "SName": "Заблокирована (доставка клиенту)",     "SCode": "05_D",     "CardGrp": "Test"   },   {     "Num": "0000000000000006",     "Rem": "Grigoryev Denis",     "SName": "В работе",     "SCode": "00",     "CardGrp": "Test"   } ]</pre>
Описание полей	
Num	Номер карты (уникальный идентификатор)
Rem	Примечание
SName	Статус. Изменение статуса возможно только из статусов: 00 и 05_C. В остальных случаях сервис вернет ошибку. Возможные значения: В работе - 00 Заблокирована (Клиент) - 05_C Заблокирована (доставка клиенту) - 05_D Заблокирована (СБ) - 05_S Закрыта - 05_XXX
SCode	Код статуса
CardGrp	Список групп, в которых присутствует карта

Таблица 6. Структура ответа на запрос списка карт по договору

	ТР _____	
--	----------	--

## 2.4.2 Лимиты (количественные ограничители)

### 2.4.2.1 Получение списка лимитов карты

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetCardLimits*

Адрес метода: */api/emv/v2/GetCardLimits?u={u}&contract={contract}&card={card}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
card	Да	Номер карты (уникальный идентификатор)
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json;

Таблица 7. Описание входящих параметров запроса для получения списка лимитов

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;ArrayOfLimit&gt;   &lt;Limit Code="TEMPL_RLI_CURR1" Currency="V" GCat="g98" GFlag="G" Prd="F" Val="23" CurValue="0" MaxNumber="10" CurNumber="10" FlagRead="N" /&gt;   &lt;Limit Code="TEMPL_RLI_ITEM2" Currency="C" GCat="SERVICE" GFlag="C" Prd="F" Val="10001" CurValue="0" MaxNumber="10" CurNumber="10" FlagRead="N" /&gt; &lt;/ArrayOfLimit&gt;</pre>
JSON	<pre>[   {     "Code": "TEMPL_RLI_CURR1",     "Currency": "V",     "GCat": "g98",     "GFlag": "G",     "Prd": "F",     "Val": 23.0,     "CurValue": 0.0,     "MaxNumber": 10.0,     "CurNumber": 10.0,     "FlagRead": "N"   },   {     "Code": "TEMPL_RLI_ITEM2",     "Currency": "V",     "GCat": "SERVICE",     "GFlag": "C",     "Prd": "F",     "Val": 10001.0,     "CurValue": 0.0,     "MaxNumber": 10.0,     "CurNumber": 10.0,     "FlagRead": "N"   } ]</pre>
Описание полей	
Code	Код лимита (уникальный идентификатор в пределах карты)
Currency	Валюта лимита. С - валюта, V – литры (только для топлива).
GCat	Код категории или товара (пустое значение – лимит назначен на все услуги)
GFlag	G - товар, C - категория товаров, A - все категории

	TP _____	
--	----------	--

Prd	Период сброса лимита к начальному значению: F – день; F7 – неделя (пн-вс); M – месяц; Q – квартал; N – бессрочный (невозобновляемый);
Val	Размер лимита
CurValue	Доступно по лимиту (размер минус выборка по лимиту)
MaxNumber	Максимальное количество транзакций. 0 – Не установлено.
CurNumber	Доступное количество транзакций. В случае, если значение поля MaxNumber равно нулю, то значение поля CurNumber не имеет смысла.
FlagRead	Флаг: Y – Изменение лимита/ограничителя запрещено. N – Изменение лимита/ограничителя разрешено.

Таблица 8. Описание структуры ответа на запрос получения списка лимитов карты

### 2.4.2.2 Создание лимита

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: CreateCardLimit

Адрес метода: /api/emv/v1/CreateCardLimit/

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)
Prd	Нет	Период сброса лимита к начальному значению (по московскому времени): F – день; F7 – неделя (пн-вс); M – месяц; Q – квартал; N – бессрочный (невозобновляемый);
Currency	Да	Валюта лимита. C - валюта, V – литры (только для топлива).
GCode	Нет	Код товара/категории товаров (коды можно получить из метода /api/emv/v1/GetGoodsList ). Пустое поле допускается только при GFlag=A – назначение лимита на все товары. Можно объединять несколько товаров или категорий (не смешивая товары с категориями) для создания общего лимита. Указывать через запятую.
GFlag	Да	G - товар, C - категория товаров, A - все категории
Val	Да	Начальное значение лимита

Таблица 9. Описание входящих параметров запроса на получение списка лимитов

Возвращаемый результат содержит следующую структуру:

- RequestId – id заявки в ЛК
- LimitCode – код лимита

Пример вызова на javascript:

```
function CreateCardLimit() {
    var reqParam = {
        "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
        "Currency": "V",
        "GCode": "FUEL",
        "Prd": "F",
        "GFlag": "C",
        "Val": 5000
    };

    var xx = $.ajax({
        type: 'POST',
        headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
        url: '/api/emv/v1/CreateCardLimit',
    });
}
```

	2.0.0.32	20 / 65
--	----------	---------

	TP _____	
--	----------	--

```

data: reqParam,
success: function (e) {
},
error: function (e) {
}
});
}

```

### 2.4.2.3 Изменение лимита

Если у лимита в поле «FlagRead» установлено значение «Y», то изменить такой лимит нельзя.

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: EditCardLimit

Адрес метода: /api/emv/v1/EditCardLimit/

Параметр	Обяз.	Значение	Комментарий
U	Да	Логин от ЛК	
Contract	Да	Код договора	
Card	Да	Графический номер карты	
LimitCode	Да	Код лимита	
Val	Да	Новое значение лимита	
CurValue	Нет	Текущая величина лимита	Заполнять в том случае, если необходимо сохранять значения в истории заявок ЛК. Справочная информация!
Prd	Нет	Период сброса лимита к начальному значению: F – день; F7 – неделя (пн-вс); М – месяц; Q – квартал; N – бессрочный (невозобновляемый);	
Currency	Нет	Валюта лимита. С - валюта, V – литры (только для топлива).	
GCode	Нет	Код(ы) товара/категории товаров	
GFlag	Нет	G - товар, С - категория товаров, А - все категории	

Таблица 10. Описание входящих параметров запроса на изменение лимита

Возвращаемый результат содержит следующую структуру:

- RequestId – id заявки в ЛК
- LimitCode – код лимита

Пример вызова на javascript:

```

function EditLimit() {
var reqParam = {
  "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
  "LimitCode": "TEMPL_RLI_ITEM63",
  "Val": 10001
};
var xx = $.ajax({
  type: 'POST',
  headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
  url: '/api/emv/v1/EditCardLimit',
  data: reqParam,
  success: function (text) {
  },
  error: function (text) {
  }
});
}

```

#### 2.4.2.4 Удаление лимита

Для удаления достаточно передать код лимита. ВНИМАНИЕ! Операция возврата учитывается в тех же лимитах, где была учтена операция обслуживания. Если лимит удалить и создать новый, то возвраты (сделанные после удаления лимита) по операциям связанных с удалённым лимитом не изменят новый лимит.

Если у лимита в поле «FlagRead» установлено значение «Y», то удалить такой лимит нельзя.

##### Метод и его входные параметры

HTTP-метод: POST  
Имя метода: DeleteCardLimit  
Адрес метода: /api/emv/v1/DeleteCardLimit/

Параметр	Обяз.	Значение	Комментарий
U	Да	Логин от ЛК	
Contract	Да	Код договора	
Card	Да	Графический номер карты	
LimitCode	Да	Код лимита	
CurValue	Нет	Текущая величина лимита	Заполнять в том случае, если необходимо сохранять значения в истории заявок.
Prd	Нет	Период сброса лимита к начальному значению: F – день; F7 – неделя (пн-вс); М – месяц; Q – квартал; N – бессрочный (невозобновляемый);	
Currency	Нет	С - валюта, V – литры	
GCode	Нет	Код товара/категории товаров	
GFlag	Нет	G - товар, С - категория товаров, А - все категории	
Val	Нет	Начальное значение лимита	

Таблица 11. Описание входящих параметров запроса на удаление лимита

Возвращаемый результат содержит следующую структуру:

- RequestID – id заявки в ЛК
- LimitCode – не используется

##### Пример вызова на javascript:

```
function DeleteLimit() {
    var reqParam = {
        "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
        "limitCode": "TEMPL_RLI_ITEM6",
    };
    var xx = $.ajax({
        type: 'POST',
        headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
        url: '/api/emv/v1/DeleteCardLimit',
        data: reqParam,
        success: function (e) {
        },
        error: function (e) {
        }
    });
}
```

#### 2.4.2.5 Удаление всех лимитов карты

Метод удаляет с карты все разрешенные для изменения лимиты (количественные ограничители) включая ограничители по времени.

Т.е. если у лимита в поле «FlagRead» установлено значение «Y», то такой лимит не будет удалён.

##### Метод и его входные параметры

HTTP-метод: POST  
Имя метода: DeleteCardLimitAll  
Адрес метода: /api/emv/v2/DeleteCardLimitAll /

	ТР _____	
--	----------	--

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)

Описание входящих параметров запроса на удаление всех лимитов карты

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RequestStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11669816" Code="0" /&gt;</pre>
JSON	<pre>{   "RequestID": 11669816.0,   "Code": "0" }</pre>
Описание полей	
RequestID	ID заявки
Code	Код ошибки

Описание структуры ответа на запрос удаления всех лимитов карты

## 2.4.3 Товарные ограничители

### 2.4.3.1 Получение списка товарных ограничителей карты

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetGoodsCardRestrictions*

Адрес метода: */api/emv/v1/GetGoodsCardRestrictions?u={u}&contract={contract}&card={card}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
card	Да	Номер карты
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

Таблица 12. Описание входящих параметров запроса на товарные ограничители карты

#### Возвращаемый результат

Формат	Результат
XML	<pre>&lt;ArrayOfRestriction&gt;   &lt;Restriction GCat="gMasla" GFlag="G" Id="229838" Type="E" FlagRead="N" /&gt;   &lt;Restriction GCat="GOODS" GFlag="C" Id="229840" Type="E" FlagRead="N" /&gt; &lt;/ArrayOfRestriction&gt;</pre>
JSON	<pre>[   {     "GCat": "gMasla",     "GFlag": "G",     "Id": "229838",     "Type": "E",     "FlagRead": "N"   },   {     "GCat": "GOODS",     "GFlag": "C",</pre>

	TP _____	
--	----------	--

	<pre>"Id": "229840", "Type": "E", "FlagRead": "N" }]</pre>
<b>Описание полей</b>	
GCat	Код товара/услуги или категории товаров
GFlag	G – товар/услуга, C - категория товаров, A - все категории
Id	Идентификатор запрета
Type	Тип ограничителя. Логика товарных ограничителей строится на разрешениях. Возможные значения: <ul style="list-style-type: none"> <li>• E – разрешено;</li> <li>• D – запрещено;</li> <li>• R – условно разрешено (не используется)</li> </ul>
FlagRead	Флаг: Y – Изменение ограничителя запрещено. N – Изменение ограничителя разрешено.

Таблица 13. Описание результата запроса товарных ограничителей карты



	ТР _____	
--	----------	--

### 2.4.3.2 Создание товарного ограничителя карты

#### Метод и его входные параметры

HTTP-метод: *POST*

Имя метода: *CreateGoodsCardRestriction*

Адрес метода: */api/emv/v1/CreateGoodsCardRestriction/*

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты
GCode	Да	Код товара / Код категории товаров (коды можно получить из метода <i>/api/emv/v1/GetGoodsList</i> ).
GFlag	Да	G – товар/услуга, C - категория товаров.
RType	Нет	E – разрешено (по умолчанию); D – запрещено

Таблица 14. Описание входящих параметров запроса создания товарного ограничителя

Результат содержит следующую структуру

- RequestId – id заявки в ЛК
- RestrictionId – id товарного запрета

Если разрешить какой либо вид товара или категорию, то автоматически будут запрещены все другие неразрешённые.

Пример. Разрешено : ДТ, Услуги.

Категория	Товар/Услуга	Обслуживание
Товары	AdBlue	Нет
Товары	Автоаксессуары	Нет
Товары	Кафе	Нет
Товары	Масла	Нет
Товары	Сопутствующий товар	Нет
Товары	Табак	Нет
Топливо	Бензин АИ-80-K5	Нет
Топливо	Бензин АИ-92-K5	Нет
Топливо	Бензин АИ-92-K5 - Фирменное	Нет
Топливо	Бензин АИ-95-K5	Нет
Топливо	Бензин АИ-95-K5 - Фирменное	Нет
Топливо	Бензин АИ-98-K5	Нет
Топливо	Бензин АИ-98-K5 - Фирменное	Нет
Топливо	Дизельное топливо	Да
Топливо	Дизельное топливо арктическое	Нет
Топливо	Дизельное топливо зимнее	Нет
Топливо	Дизельное топливо межсезонное	Нет
Топливо	Дизельное топливо Фирменное	Нет
Топливо	Метан	Нет
Топливо	Пропан/бутан	Нет
Топливо	Топливо моторное среднестиллятное сорт 4	Нет
Топливо	Топливо печное бытовое вид IV	Нет
Услуги	Гостиница	Да
Услуги	Мойка	Да
Услуги	Оплата дорог	Да
Услуги	Прочие услуги	Да
Услуги	Пылесос	Да
Услуги	Стоянка	Да
Услуги	Шиномонтаж	Да

Пример вызова на javascript:

```
function CreateGoodsRestriction() {
    var reqParam = {
        "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
        "GCode": "SERVICE",
        "GFlag": "C",
        "RType": "E"
    };
    var xx = $.ajax({
        type: 'POST',
        headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
        url: '/api/emv/v1/CreateGoodsCardRestriction',
        data: reqParam,
        success: function (text) { },
        error: function (text) { }
    });
}
```

### 2.4.3.3 Удаление товарного ограничителя

Метод позволяет удалить товарный ограничитель. Если у ограничителя в поле «FlagRead» установлено значение «Y», то удалить такой ограничитель нельзя.

#### Метод и его входные параметры

HTTP-метод: *POST*

Имя метода: *DeleteGoodsRestriction*

Адрес метода: */api/emv/v1/DeleteGoodsRestriction*

Параметр	Обяз.	Значение	
U	Да	Логин от ЛК	
Contract	Да	Код договора	
Card	Да	Графический номер карты	
Id	Да	Id запрета	
GCode	Нет	Код товара / Код категории товаров (коды можно получить из метода /api/emv/v1/GetGoodsList). Можно объединять несколько товаров или категорий (не смешивая товары с категориями) для создания общего товарного ограничения. Указывать через запятую.	Заполнять в случае необходимости сохранять значения в журнале изменения товарного ограничения.
GFlag	Нет	G – товар/услуга, C - категория товаров.	
RType	Нет	E – разрешено; D – запрещено	

Таблица 15. Описание входящих параметров запроса удаления товарного ограничителя

Результат содержит следующую структуру

- RequestId – id заявки в ЛК
- RestrictionId – не используется

```
function DeleteGoodsRestriction() {
    var reqParam = {
        "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
        "Id": "1179"
    };
    var xx = $.ajax({
        type: 'POST',
        headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
        url: '/api/emv/v1/DeleteGoodsRestriction',
        data: reqParam,
        success: function (text) {},
        error: function (text) {}
    });
}
```

	ТР _____	
--	----------	--

#### 2.4.3.4 Удаление всех товарных ограничителей карты

Метод удаляет с карты все разрешенные для изменения товарные ограничители.

Т.е. если у ограничителя в поле «FlagRead» установлено значение «Y», то такой ограничитель не будет удалён.

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: DeleteCardGoodsRestrictionAll

Адрес метода: /api/emv/v2/DeleteCardGoodsRestrictionAll/

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)

Описание входящих параметров запроса на удаление всех товарных ограничителей карты

##### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RequestStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11669816" Code="0" /&gt;</pre>
JSON	<pre>{   "RequestID": 11669816.0,   "Code": "0" }</pre>
Описание полей	
RequestID	ID заявки
Code	Код ошибки

Описание структуры ответа на запрос удаления всех товарных ограничителей карты

#### 2.4.3.5 Изменение товарного ограничителя

Метод позволяет изменить товарный ограничитель. Если у ограничителя в поле «FlagRead» установлено значение «Y», то изменить такой ограничитель нельзя.

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: EditGoodsRestriction

Адрес метода: /api/emv/v1/EditGoodsRestriction

Параметр	Обяз.	Значение	
U	Да	Логин от ЛК	
Contract	Да	Код договора	
Card	Да	Графический номер карты	
Id	Да	Id запрета	
NewGCode	Да	Новый код товара / Новый код категории товаров (коды можно получить из метода /api/emv/v1/GetGoodsList).	
NewGFlag	Да	Тип нового кода товара/категории: G – товар,	

	2.0.0.32	27 / 65
--	----------	---------

	ТР _____	
--	----------	--

		С - категория товаров	
NewRType	Нет	Новый тип товарного ограничителя: Е – разрешено (по умолчанию), D – запрещено.	
GCode	Нет	Код товара / Код категории товаров (коды можно получить из метода /api/emv/v1/GetGoodsList). Можно объединять несколько товаров или категорий (не смешивая товары с категориями) для создания общего товарного ограничения. Указывать через запятую.	Заполнять в случае необходимости сохранять значения в журнале изменения товарного ограничения.
GFlag	Нет	G – товар/услуга, С - категория товаров.	
RType	Нет	E – разрешено, D – запрещено	

*Описание входящих параметров запроса изменения товарного ограничителя*

Результат содержит следующую структуру

- RequestId – id заявки в ЛК
- RestrictionId – не используется

```
function EditGoodsRestriction() {
    var reqParam = {
        "U": "u", "Contract": "ISS01TEST", "Card": "0000000000000000",
        "Id": "1179",
        "GCode": "SERVICE",
        "GFlag": "C",
        "RType": "E"
        "NewGCode": "gmoika",
        "NewGFlag": "G",
        "NewRType": "E"
    };
    var xx = $.ajax({
        type: 'POST',
        headers: { "Accept": "application/json", "RnCard-Identity-Account-Pass": "ПАРОЛЬBASE64" },
        url: '/api/emv/v1/EditGoodsRestriction',
        data: reqParam,
        success: function (text) {},
        error: function (text) {}
    });
}
```

	ТР _____	
--	----------	--

## 2.4.4 Блокировка карт

### Метод и его входные параметры

HTTP-метод: POST

Имя метода: BlockingCard

Адрес метода: /api/emv/v1/BlockingCard

HTTP-метод	Тип	Обяз.	POST
u	string	Да	Логин от ЛК
contract	string	Да	Код договора
card	string	Да	Номер карты

Таблица 16. Описание входящих параметров метода блокировки карты

### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат	
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ResponseBlockCard xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;RequestID&gt;16052783&lt;/RequestID&gt;   &lt;Code&gt;1&lt;/Code&gt; &lt;/ResponseBlockCard&gt;</pre>	
JSON	<pre>{   "RequestID": 16052783.0,   "Code": 1.0 }</pre>	
Описание полей		
Параметр	Тип	Описание
RequestID	decimal	ID заявки в ЛК
Code	decimal	Код. В настоящее время не используется.

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

### Пример запроса(CURL):

Блокировка карты	
Запрос	<pre>curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&amp;Contract=ISS01TEST&amp;Card=782601000000238' https://lkapi.rn-card.ru/api/emv/v1/BlockingCard</pre>
Ответ	<pre>{   "RequestID": 16052783.0,   "Code": 1.0 }</pre>

## 2.4.5 Разблокировка карт

### Метод и его входные параметры

HTTP-метод: POST

Имя метода: UnblockCard

Адрес метода: /api/emv/v1/UnblockingCard

Параметр	Тип	Обяз.	Описание
u	string	Да	Логин от ЛК
contract	string	Да	Код договора
card	string	Да	Графический номер карты

	2.0.0.32	29 / 65
--	----------	---------

	ТР _____	
--	----------	--

Таблица 17. Описание входящих параметров метода разблокировки карты

#### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат	
XML	<?xml version="1.0" encoding="utf-8"?> <ResponseBlockCard xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <RequestID>16052691</RequestID> <Code>1</Code> </ResponseBlockCard>	
JSON	{ "RequestID": 16052691.0, "Code": 1.0 }	
Описание полей		
Параметр	Тип	Описание
RequestID	decimal	ID заявки в ЛК
Code	decimal	Код. В настоящее время не используется.

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

#### Пример запроса(CURL):

Создание группы карт	
Запрос	<pre>curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&amp;Contract=ISS01TEST&amp;Card=7826010000000238' https://lkapi.rn-card.ru/api/emv/v1/UnblockingCard</pre>
Ответ	<pre>{   "RequestID": 16052691.0,   "Code": 1.0 }</pre>

## 2.4.6 Получение информации о статусе карт

Метод возвращает статус карт.

#### Метод и его входные параметры

HTTP-метод	POST		
Имя метода	InfoCardStatus		
Адрес метода	https://lkapi.rn-card.ru/api/emv/v2/InfoCardStatus		
Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин от ЛК
Contract	string	Да	Код договора
CardList[]	object	Усл.	Массив карт. Массив содержит графические номера карт. (Если не задан, то будут возвращены все карты по договору)

Таблица 18. Описание входящих параметров метода управления группами карт

#### Возвращаемый результат:

Результат будет содержать следующую структуру:

Формат результата	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;InfoCardList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;CardList&gt;     &lt;Card Num="7826010000000238" Rem="Прокофьев Вадим Сергеевич; suzuki: 1111" SName="В работе" SCode="00" /&gt;   &lt;/CardList&gt; &lt;/InfoCardList&gt;</pre>

	2.0.0.32	30 / 65
--	----------	---------

	TP _____	
--	----------	--

	<pre> &lt;Card Num="7826010000000246" Rem="" SName="Заблокирована (Клиент)" SCode="05_C" /&gt; &lt;/CardList&gt; &lt;/InfoCardList&gt; </pre>
JSON	<pre> {   "CardList": [{     "Num": "7826010000000238",     "Rem": "Прокофьев Вадим Сергеевич; suzuki: 1111",     "SName": "В работе",     "SCode": "00"   },   {     "Num": "7826010000000246",     "Rem": "",     "SName": "Заблокирована (Клиент)",     "SCode": "05_C"   } ] } </pre>

#### Примеры запросов(CURL):

Запрос	<pre> curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard- Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&amp; Contract=ISS01TEST&amp;CardList[0]=7826010000000238&amp;CardList[1]=7826010000000246' https://lkapi.rn-card.ru/api/emv/v2/InfoCardStatus </pre>
Ответ	<pre> {   "CardList": [{     "Num": "7826010000000238",     "Rem": "Прокофьев Вадим Сергеевич; suzuki: 1111",     "SName": "В работе",     "SCode": "00"   },   {     "Num": "7826010000000246",     "Rem": "",     "SName": "Заблокирована (Клиент)",     "SCode": "05_C"   } ] } </pre>

## 2.4.7 Количество попыток ввода пин-кода карты

### 2.4.7.1 Получение информации о количестве попыток ввода пин-кода

#### Метод и его входные параметры

HTTP-метод: GET

Имя метода: InfoCardPin

Адрес метода: /api/emv/v2/InfoCardPin?u={u}&contract={contract}&card={card}&type={type}

HTTP-метод	Тип	Обяз.	GET
u	string	Да	Логин от ЛК
contract	string	Да	Код договора
card	string	Да	Номер карты
type	string	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

Таблица 19. Описание входящих параметров метода блокировки карты

#### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

	2.0.0.32	31 / 65
--	----------	---------

	TP _____	
--	----------	--

Формат	Результат	
XML	<pre>&lt;?xml version="1.0" encoding="UTF-8"?&gt; &lt;CardPinInfo xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Card="7826010700000000137" Max="3" Current="0" /&gt;</pre>	
JSON	<pre>{   "Card": "7826010700000000137",   "Max": 3.0,   "Current": 0.0, }</pre>	
Описание полей		
Параметр	Тип	Описание
Card	string	Номер карты
Max	decimal	Максимальное количество попыток ввода пин-кода
Current	decimal	Текущее количество попыток ввода пин-кода

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

#### Пример запроса(CURL):

Сброс количества попыток ввода пин-кода карты	
Запрос	<pre>curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -G -d 'U=demo&amp;Contract=ISS01TEST&amp;Card=78260100000000238' https://lkapi.rn-card.ru/api/emv/v2/InfoCardPin</pre>
Ответ	<pre>{   "Card": "78260100000000238",   "Max": 3.0,   "Current": 0.0 }</pre>

#### 2.4.7.2 Сброс количества попыток ввода пин-кода карты

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: ClearCardPin

Адрес метода: /api/emv/v2/ClearCardPin

HTTP-метод	Тип	Обяз.	POST
u	string	Да	Логин от ЛК
contract	string	Да	Код договора
card	string	Да	Номер карты

Таблица 20. Описание входящих параметров метода блокировки карты

##### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="UTF-8"?&gt; &lt;RequestStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11604200" Code="0" /&gt;</pre>
JSON	<pre>{   "RequestID": "11604200.0",   "Code": "0" }</pre>
Описание полей	



	ТР _____	
--	----------	--

Параметр	Тип	Описание
RequestID	decimal	ID заявки в ЛК
Code	string	Код ошибки. Соответствует коду ошибки из п.2.2. В настоящее время не используется.

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

#### Пример запроса(CURL):

Сброс количества попыток ввода пин-кода карты	
Запрос	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&Card=782601000000238' https://lkapi.rn-card.ru/api/emv/v2/ClearCardPin
Ответ	{ "RequestID": 11604200.0, "Code": "0" }

## 2.4.8 Виртуальная карта

### 2.4.8.1 Закрытие виртуальной карты

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: RequestCardClose

Адрес метода: /api/emv/v2/RequestCardClose

HTTP-метод	Тип	Обяз.	POST
U	string	Да	Логин от ЛК
Contract	string	Да	Код договора
Card	string	Да	Номер карты

Таблица 21. Описание входящих параметров метода блокировки карты

#### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат		Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RequestStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11674890" Code="0" /&gt;</pre>	
JSON	<pre>{   "RequestID": 11674890.0,   "Code": "0" }</pre>	
Описание полей		
Параметр	Тип	Описание
RequestID	decimal	ID заявки в ЛК
Code	string	Код. В настоящее время не используется.

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

#### Пример запроса(CURL):

Закрытие виртуальной карты	
Запрос	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d

	2.0.0.32	33 / 65
--	----------	---------

	TP _____	
--	----------	--

	'U=demo&Contract=ISS01TEST&Card=7826010790000001920' https://lkapi.rn-card.ru/api/emv/v2/RequestCardClose
Ответ	{ "RequestID": 11674890.0, "Code": "0" }

### 2.4.8.2 Создание виртуальной карты

Метод, позволяет создать заявку на создание одной или нескольких виртуальных карт для использования в мобильном приложении РН-Карт

#### Метод и его входные параметры

HTTP-метод: *POST*

Имя метода: *VirtualCard*

Адрес метода: */api/emv/v1/VirtualCard*

Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин
Contract	string	Да	Код договора
Cards	Object[]	Да	Массив объектов. См. описание объекта «Информация о картах»
Объект: Информация о картах			
LName	string	Да	Фамилия
FName	string	Да	Имя
MName	string	Нет	Отчество
Tel	string	Да	Номер телефона. На данный номер будет отправлен код активации карты. Формат: +79191112233 Возможно понадобится заменить + на utf8 код %2B. %2B79191112233
EDate	string	Нет	Дата окончания срока действия карты. Максимальная дата срока действия карты – 50 лет от месяца и года создания заявки. Если дата не передана в качестве параметра, то дата окончания срока действия карты будет установлена как дата создания заявки + 30 лет. Формат: «ММYY», где ММ – две цифры месяца, YY две цифры года. Например: январь 2018 года – 0118 (необязательный параметр)

#### Возвращаемый результат:

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру

Формат	Результат
XML	<?xml version="1.0" encoding="utf-8"?> <ResponseVirtualCard xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><RequestID>13183219</RequestID></ResponseVirtualCard>
JSON	{ "RequestID":13183219.0 }

#### Описание полей возвращаемого результата:

Параметр	Тип	Описание
RequestID	decimal	ID созданной заявки. При формировании заявки, на каждую запись с информацией о создаваемой виртуальной карте создаётся отдельная заявка. Первая запись считается головной заявкой, остальные считаются подчинёнными. У подчинённых заявок в поле ParentID устанавливается значение RequestID из головной заявки.

	2.0.0.32	34 / 65
--	----------	---------

	ТР _____	
--	----------	--

В случае наличия сообщения об отказе формируется стандартное сообщение с кодом ошибки HTTP-сервера

Код ошибки	Сообщение об ошибке	Описание
400	Bad Request	Критическая ошибка
403	Access forbidden	Доступ запрещён
404	Not found	Данные не найдены

### Пример запроса

#### CURL:

```
curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&Cards[0][LName]=pupkin&Cards[0][FName]=vasya&Cards[0][MName]=v&Cards[0][Tel]=%2B79169163311&Cards[0][EDate]=0318&Cards[1][LName]=petrov&Cards[1][FName]=nikolay&Cards[1][Tel]=%2B79169163311' https://lkapi.rn-card.ru/api/emv/v1/VirtualCard
```

### 2.4.8.3 Получение информации о заявках на создание виртуальных карт

Метод, позволяет получить информацию о заявках на создание виртуальных карт.

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *VirtualCard*

Адрес метода: */api/emv/v1/VirtualCard*

Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин
Contract	string	Да	Код договора
RequestID	decimal	Нет	ID заявки. Если параметр указан, то возвращается информация по ID заявки, включая информацию о подчиненных/вложенных заявках
BeginDate	DateTime	Нет	Начальная дата выборки информации о созданных заявках. Если дата не задана, то выборка делается с начала текущего месяца
EndDate	DateTime	Нет	Конечная дата выборки информации о созданных заявках. Если дата не задана, то выборка делается по текущую дату.
Type	string	Нет	Тип возвращаемого результата: XML или JSON

#### Возвращаемый результат:

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;HistoryVirtualCardList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;RequestList&gt;     &lt;Item RequestID="11603847" RequestDate="2018-01-18T14:52:34" RequestStatus="N" LastName="pupkin" FirstName="vasya" MiddleName="v" PhoneNum="+79169163311" EndDate="0318" CardNum="" ParentID="11603847" /&gt;     &lt;Item RequestID="11603848" RequestDate="2018-01-18T14:52:34" RequestStatus="N" LastName="petrov" FirstName="nikolay" MiddleName="" PhoneNum="+79169163311" EndDate="" CardNum="" ParentID="11603847" /&gt;   &lt;/RequestList&gt; &lt;/HistoryVirtualCardList&gt;</pre>
JSON	{

	2.0.0.32	35 / 65
--	----------	---------

```

"RequestList": [
  {
    "RequestID": 11603847,
    "RequestDate": "2018-01-18T14:52:34",
    "RequestStatus": "N",
    "LastName": "pupkin",
    "FirstName": "vasya",
    "MiddleName": "v",
    "PhoneNum": "+79169163311",
    "EndDate": "0318",
    "CardNum": "",
    "ParentID": 11603847
  },
  {
    "RequestID": 11603848,
    "RequestDate": "2018-01-18T14:52:34",
    "RequestStatus": "N",
    "LastName": "petrov",
    "FirstName": "nikolay",
    "MiddleName": "",
    "PhoneNum": "+79169163311",
    "EndDate": "",
    "CardNum": "",
    "ParentID": 11603847
  }
]

```

#### Описание полей возвращаемого результата:

Описание полей возвращаемого результата		
Имя	Тип	Описание
RequestList	Object[]	Массив объектов.
Описание полей объекта RequestList		
RequestID	decimal	ID созданной заявки
RequestDate	DateTime	Дата создания заявки
RequestStatus	string	Статус заявки. P – выполнена, S – остановлена, N – новая
LastName	string	Фамилия
FirstName	string	Имя
MiddleName	string	Отчество
PhoneNum	string	Номер мобильного телефона
EndDate	string	Дата окончания срока действия карты. Формат: «ММYY», где ММ – две цифры месяца, YY две цифры года. Например: январь 2018 года – 0118
CardNum	string	Номер созданной виртуальной карты.
ParentID	decimal	Номер головной заявки. При формировании заявки, на каждую запись с информацией о создаваемой виртуальной карте создаётся отдельная заявка. Первая запись считается головной заявкой, остальные считаются подчинёнными. У подчинённых заявок в поле ParentID устанавливается значение Request ID из головной заявки.

#### Пример запроса

#### CURL:

*curl -X GET 'https://lkapi.rn-card.ru/api/emv/v1/virtualcard/?u=demo&contract=ISS01TEST'*

	ТР _____	
--	----------	--

#### 2.4.8.4 Получение информации о виртуальной карте

Метод, позволяет получить информацию о виртуальной карте.

##### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *InfoCardVirtual*

Адрес метода: */api/emv/v2/InfoCardVirtual*

Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин
Card	string	Да	Графический номер карты
Type	string	Нет	Тип возвращаемого результата: XML или JSON

##### Возвращаемый результат:

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;VirtualCardInfo xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Card="782601070000000001" Fio="Иванов Иван Иванович" Phone="+79161112301" /&gt;</pre>
JSON	<pre>{   "Card": "782601070000000001",   "Fio": "Иванов Иван Иванович",   "Phone": "+79161112301" }</pre>

##### Описание полей возвращаемого результата:

Описание полей возвращаемого результата		
Имя	Тип	Описание
Card	string	Графический номер карты
Fio	string	Ф. И. О.
Phone	string	Номер мобильного телефона

##### Пример запроса:

##### CURL:

```
curl -X GET -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' https://lkapi.rn-
card.ru/api/emv/v2/InfoCardVirtual/?u=demo&card=78260107000000000001
```

#### 2.4.8.5 Изменение Ф.И.О. и/или телефона у виртуальной карты

Метод, позволяет создать заявку на изменение Ф.И.О. и/или телефона у виртуальной карты

##### Метод и его входные параметры

HTTP-метод: *POST*

Имя метода: *SetVirtualCardOwner*

Адрес метода: */api/emv/v2/SetVirtualCardOwner*

Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин
Contract	string	Да	Код договора

	2.0.0.32	37 / 65
--	----------	---------

	TP _____	
--	----------	--

Card	string	Да	Графический номер карты
LastName	string	Да	Фамилия
FirstName	string	Да	Имя
MiddleName	string	Нет	Отчество
Phone	string	Да	Номер телефона. На данный номер будет отправлен код активации карты. Формат: +79191112233

#### Возвращаемый результат:

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру

Формат		Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RequestStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11674890" Code="0" /&gt;</pre>	
JSON	<pre>{   "RequestID": 11674890.0,   "Code": "0" }</pre>	
Описание полей		
Параметр	Тип	Описание
RequestID	decimal	ID заявки в ЛК
Code	string	Код. В настоящее время не используется.

#### Пример запроса

##### CURL:

```
curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАРОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&Card=78260107000000000001&LastName=%D0%98%D0%B2%D0%B0%D0%BD%D0%BE%D0%B2&FirstName=%D0%98%D0%B2%D0%B0%D0%BD&MiddleName=%D0%98%D0%B2%D0%B0%D0%BD%D0%BE%D0%B2%D0%B8%D1%87&Phone=%2B79161112345' https://lkapi.rn-card.ru/api/emv/v2/SetVirtualCardOwner
```

## 2.4.9 Региональные ограничители карты

Описание работы региональных ограничителей см. [п.2.3.2.](#)

### 2.4.9.1 Получение списка региональных ограничителей карты

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetCardRegionRestriction*

Адрес метода: */api/emv/v2/GetCardRegionRestriction?u={u}&contract={contract}&card={card}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
card	Да	Номер карты
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt;</pre>

	2.0.0.32	38 / 65
--	----------	---------

	TP _____	
--	----------	--

	<pre>&lt;RestrictionList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;Item CountryCode="RUS" RegionCode="01,11" NpoCode="" PosCode="" RestrictionType="E" RestrictionID="259483" FlagRead="N" /&gt;   &lt;Item CountryCode="" RegionCode="" NpoCode="" PosCode="AZS101309" RestrictionType="D" RestrictionID="261103" FlagRead="N" PartnerCode="" /&gt; &lt;/RestrictionList&gt;</pre>
JSON	<pre>{   "RestrictionList": [     {       "CountryCode": "RUS",       "RegionCode": "01,11",       "NpoCode": "",       "PosCode": "",       "RestrictionType": "E",       "RestrictionID": "259483",       "FlagRead": "N",       "PartnerCode": ""     },     {       "CountryCode": "",       "RegionCode": "",       "NpoCode": "",       "PosCode": "AZS101309",       "RestrictionType": "D",       "RestrictionID": "261103",       "FlagRead": "N",       "PartnerCode": ""     }   ] }</pre>
<b>Описание полей</b>	
CountryCode	Трёхбуквенный код страны. Может содержать несколько кодов указанных через разделитель «,»(запятая).
RegionCode	Код региона Может содержать несколько кодов указанных через разделитель «,»(запятая).
NpoCode	Код НПО. Может содержать несколько кодов указанных через разделитель «,»(запятая).
PosCode	Код ТО Может содержать несколько кодов указанных через разделитель «,»(запятая).
RestrictionType	Тип ограничителя. Возможные значения: E – Обслуживание разрешено, D – Обслуживание запрещено.
RestrictionID	ID ограничителя
FlagRead	Флаг: Y – Изменение ограничителя запрещено. N – Изменение ограничителя разрешено.
PartnerCode	Код партнёра

Описание структуры ответа на запрос получения списка региональных ограничителей карты

#### 2.4.9.2 Создание/Изменение/Удаление регионального ограничителя карты

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: CardRegionRestriction

Адрес метода: /api/emv/v2/CardRegionRestriction/

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора

	2.0.0.32	39 / 65
--	----------	---------

	ТР _____	
--	----------	--

Card	Да	Графический номер карты (уникальный идентификатор)
Act	Да	Действие с региональным ограничителем: С – Создать, М – Изменить, D – Удалить.
RestrictionType	Усл. обяз	Тип ограничителя. Возможные значения: Е – Обслуживание разрешено, D – Обслуживание запрещено. Параметр является обязательным при создании или изменении регионального ограничителя.
CountryCode	Нет	Трёхбуквенный код страны. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 60.
RegionCode	Нет	Код региона. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 30. В параметре CountryCode необходимо указать страны, регионы которых перечислены в RegionCode.
PartnerCode	Нет	Код партнёра. Запрещено указание межхостовых партнёров (без эквайринговых данных, тип партнёра «Н»).
PosCode	Нет	Код ТО. Может содержать несколько кодов ТО(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 200.
NpoCode	Нет	Код НПО. Может содержать несколько кодов стран(указанных через разделитель «,»(запятая)). Максимальное количество кодов в списке – 10.
RestrictionCode	Усл. обяз	Код удаляемого или изменяемого ограничителя. Параметр является обязательным только при удалении или изменении регионального ограничителя.

*Описание входящих параметров запроса на создание регионального ограничителя*

Пример ограничителя:

Act = C

RegionCode = 11 (Архангельская область)

CountryCode = RUS RestrictionType = D

С указанными параметрами будет создан региональный ограничитель, по которому будет запрещено обслуживание в Архангельской области РФ.

В случае если код региона не будет принадлежать ни одной стране, то будет выдана ошибка.

Заполнять поля RestrictionType, CountryCode, RegionCode, PartnerCode, PosCode, NpoCode при удалении ограничителя необходимо для сохранения старых значений ограничителя в журнале изменения региональных ограничителей.

При изменении ограничителя, ограничитель будет изменён в соответствии со всеми новыми значениями параметров. Если параметр оставить пустым, то у ограничителя параметр станет пустым. Например, если при изменении ограничителя «Россия/Новосибирская область» параметр RegionCode оставить пустым, то ограничитель будет изменён на «Россия».

Если у ограничителя в поле «FlagRead» установлено значение «Y», то изменить или удалить такой ограничитель нельзя.

	2.0.0.32	40 / 65
--	----------	---------



	TP _____	
--	----------	--

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RegionRestrictionResponse xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Contract="ISS001933" Card="7826011390000119" RequestID="11665096" ErrorCode="0" ErrorMessage="[0] Выполнено успешно" RestrictionID="261303" /&gt;</pre>
JSON	<pre>{   "Contract": "ISS001933",   "Card": "7826011390000119",   "RequestID": 11665096.0,   "ErrorCode": 0,   "ErrorMessage": "[0] Выполнено успешно",   "RestrictionID": "261303" }</pre>
Описание полей	
Contract	Код договора
Card	Графический номер карты
RequestID	ID заявки
ErrorCode	Код ошибки
ErrorMessage	Сообщение об ошибке
RestrictionID	ID ограничителя

*Описание структуры ответа на запрос создания регионального ограничителя карты*

#### 2.4.9.3 Удаление всех региональных ограничителей карты

Метод удаляет с карты все разрешенные для изменения региональные ограничители. Т.е. если у ограничителя в поле «FlagRead» установлено значение «Y», то такой ограничитель не будет удалён.

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: DeleteAllCardRegionRestriction

Адрес метода: /api/emv/v2/DeleteAllCardRegionRestriction /

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)

*Описание входящих параметров запроса на удаление всех региональных ограничителей карты*

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;RegionRestrictionResponse xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Contract="ISS001933" Card="7826011390000119" RequestID="11669811" ErrorCode="0" ErrorMessage="[0] Выполнено успешно" /&gt;</pre>
JSON	<pre>{   "Contract": "ISS001933",   "Card": "7826011390000119",   "RequestID": 11669811.0,   "ErrorCode": 0,   "ErrorMessage": "[0] Выполнено успешно", }</pre>

	2.0.0.32	41 / 65
--	----------	---------

	ТР _____	
--	----------	--

	<code>"RestrictionID": null</code>
	<code>}</code>
Описание полей	
Contract	Код договора
Card	Графический номер карты
RequestID	ID заявки
ErrorCode	Код ошибки
ErrorMessage	Сообщение об ошибке
RestrictionID	ID ограничителя

Описание структуры ответа на запрос удаления всех региональных ограничителей карты

## 2.4.10 Ограничители по времени для карты

### 2.4.10.1 Получение списка ограничителей по времени для карты

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetCardTimeRestriction*

Адрес метода: */api/emv/v2/GetCardTimeRestriction?u={u}&contract={contract}&card={card}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
card	Да	Номер карты
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;TimeRestrictions xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"&gt;   &lt;Item Code="TEMPL_RLI_CURR9" WeekDays="2" TimeInterval="00002359" FlagRead="N" /&gt;   &lt;Item Code="TEMPL_RLI_CURR11" WeekDays="3" TimeInterval="00001204" FlagRead="N" /&gt;   &lt;Item Code="TEMPL_RLI_CURR12" WeekDays="3" TimeInterval="15062359" FlagRead="N" /&gt; &lt;/TimeRestrictions&gt;</pre>
JSON	<pre>{   "TimeRestrictions": [     {       "Code": "TEMPL_RLI_CURR9",       "WeekDays": "2",       "TimeInterval": "00002359",       "FlagRead": "N"     },     {       "Code": "TEMPL_RLI_CURR11",       "WeekDays": "3",       "TimeInterval": "00001204",       "FlagRead": "N"     },     {       "Code": "TEMPL_RLI_CURR12",       "WeekDays": "3",       "TimeInterval": "15062359",       "FlagRead": "N"     }   ] }</pre>

	2.0.0.32	42 / 65
--	----------	---------

	TP _____	
--	----------	--

	<pre> "TimeInterval": "15062359", "FlagRead": "N"       }     ]   } </pre>
Описание полей	
Code	Код лимита (уникальный идентификатор в пределах карты)
WeekDays	День недели работы лимита (1 – Пн., 2 – Вт., 3- Ср., 4 – Чт., 5 – Пт., 6 – Сб., 7 – Вс.)
TimeInterval	Временной интервал которым ограничено обслуживание. Указывается в часах и минутах без разделителя. Например, интервал «15062359» ограничивает обслуживание с 15 часов 06 минут до 23 часов 59 минут включительно.
FlagRead	Флаг: Y – Изменение лимита/ограничителя запрещено. N – Изменение лимита/ограничителя разрешено.

Описание структуры ответа на запрос получения списка ограничителей по времени для карты

#### 2.4.10.2 Удаление ограничителя по времени для карты

Метод позволяет удалить ограничитель по времени. Если у ограничителя в поле «FlagRead» установлено значение «Y», то удалить такой ограничитель нельзя.

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: DeleteCardTimeRestriction

Адрес метода: /api/emv/v2/DeleteCardTimeRestriction

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)
Code	Да	Код ограничителя

Описание входящих параметров запроса на удаление ограничителя по времени для карты

##### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre> &lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ResponseResult xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11669816" ErrorCode="0" ErrorMessage="[0] Запрос выполнен успешно" Code="TEMPL_RLI_CURR45" /&gt; </pre>
JSON	<pre> {   "RequestID": 11669816.0,   "ErrorCode": "0",   "ErrorMessage": "[0] Запрос выполнен успешно",   "Code": "TEMPL_RLI_CURR45" } </pre>
Описание полей	
RequestID	ID заявки
ErrorCode	Код ошибки
ErrorMessage	Сообщение об ошибке
Code	Код ограничителя

	ТР _____	
--	----------	--

Описание структуры ответа на запрос удаления ограничителя по времени для карты

### 2.4.10.3 Создание ограничителя по времени для карты

#### Метод и его входные параметры

HTTP-метод: POST

Имя метода: CreateCardTimeRestriction

Адрес метода: /api/emv/v2/CreateCardTimeRestriction

Параметр	Обяз.	Значение
U	Да	Логин от ЛК
Contract	Да	Код договора
Card	Да	Графический номер карты (уникальный идентификатор)
WeekDays	Да	День недели работы лимита (1 – Пн., 2 – Вт., 3- Ср., 4 – Чт., 5 – Пт., 6 – Сб., 7 – Вс.)
TimeInterval	Да	Временной интервал которым ограничено обслуживание. Указывается в часах и минутах без разделителя. Например, интервал «15062359» ограничивает обслуживание с 15 часов 06 минут до 23 часов 59 минут включительно.

Описание входящих параметров запроса на удаление ограничителя по времени для карты

Временные интервалы задаются инвертированными. Т.е. ограничитель фактически блокирует карту в периоде его временных настроек, в остальные периоды лимит не оказывает влияние на авторизации.

Например, если пользователю требуется разрешить работу карты только по понедельникам с 12 до 13 и с 15 до 16 часов, по средам с 18 до 19 часов и по пятницам, то требуется создать ограничители со следующими временными параметрами:

- 1) WeekDays = 1, TimeInterval = 16011159
- 2) WeekDays = 1, TimeInterval = 13011459
- 3) WeekDays = 2, TimeInterval = 00002359
- 4) WeekDays = 3, TimeInterval = 19011759
- 5) WeekDays = 4, TimeInterval = 00002359
- 6) WeekDays = 6, TimeInterval = 00002359
- 7) WeekDays = 7, TimeInterval = 00002359

При создании ограничителя необходимо убедиться в том, что по указанному дню недели отсутствует пересечение временных периодов.

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ResponseResult xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11669816" ErrorCode="0" ErrorMessage="[0] Запрос выполнен успешно" Code="TEMPL_RLI_CURR45" /&gt;</pre>
JSON	<pre>{   "RequestID": 11669816.0,   "ErrorCode": "0",   "ErrorMessage": "[0] Запрос выполнен успешно",   "Code": "TEMPL_RLI_CURR45" }</pre>
Описание полей	

	2.0.0.32	44 / 65
--	----------	---------

	TP _____	
--	----------	--

RequestID	ID заявки
ErrorCode	Код ошибки
ErrorMessage	Сообщение об ошибке
Code	Код ограничителя

*Описание структуры ответа на запрос удаления ограничителя по времени для карты*

#### 2.4.10.4 Изменение ограничителя по времени для карты

Метод позволяет изменить ограничитель по времени. Если у ограничителя в поле «FlagRead» установлено значение «Y», то изменить такой ограничитель нельзя.

##### Метод и его входные параметры

HTTP-метод: POST

Имя метода: EditCardTimeRestriction

Адрес метода: /api/emv/v2/EditCardTimeRestriction

Параметр	Обяз.	Значение	
U	Да	Логин от ЛК	
Contract	Да	Код договора	
Card	Да	Графический номер карты (уникальный идентификатор)	
Code	Да	Код ограничителя	
WeekDays	Да	День недели работы лимита (1 – Пн., 2 – Вт., 3- Ср., 4 – Чт., 5 – Пт., 6 – Сб., 7 – Вс.)	Заполнять в случае необходимости сохранять значения в журнале изменения ограничителя по времени.
TimeInterval	Да	Временной интервал которым ограничено обслуживание. Указывается в часах и минутах без разделителя. Например, интервал «15062359» ограничивает обслуживание с 15 часов 06 минут до 23 часов 59 минут включительно.	
WeekDaysNew	Да	День недели работы лимита (1 – Пн., 2 – Вт., 3- Ср., 4 – Чт., 5 – Пт., 6 – Сб., 7 – Вс.)	
TimeIntervalNew		Временной интервал которым ограничено обслуживание. Указывается в часах и минутах без разделителя. Например, интервал «15062359» ограничивает обслуживание с 15 часов 06 минут до 23 часов 59 минут включительно.	

*Описание входящих параметров запроса на удаление ограничителя по времени для карты*

Временные интервалы задаются инвертированными. Т.е. ограничитель фактически блокирует карту в периоде его временных настроек, в остальные периоды лимит не оказывает влияние на авторизации.

Например, если пользователю требуется разрешить работу карты только по понедельникам с 12 до 13 и с 15 до 16 часов, по средам с 18 до 19 часов и по пятницам, то требуется создать ограничители со следующими временными параметрами:

- 1) WeekDays = 1, TimeInterval = 16011159
- 2) WeekDays = 1, TimeInterval = 13011459
- 3) WeekDays = 2, TimeInterval = 00002359
- 4) WeekDays = 3, TimeInterval = 19011759
- 5) WeekDays = 4, TimeInterval = 00002359
- 6) WeekDays = 6, TimeInterval = 00002359

	2.0.0.32	45 / 65
--	----------	---------

	TP _____	
--	----------	--

7) WeekDays = 7, TimeInterval = 00002359

При изменении ограничителя необходимо убедиться в том, что по указанному дню недели отсутствует пересечение временных периодов.

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ResponseResult xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" RequestID="11669816" ErrorCode="0" ErrorMessage="[0] Запрос выполнен успешно" Code="TEMPL_RLI_CURR45" /&gt;</pre>
JSON	<pre>{   "RequestID": 11669816.0,   "ErrorCode": "0",   "ErrorMessage": "[0] Запрос выполнен успешно",   "Code": "TEMPL_RLI_CURR45" }</pre>
Описание полей	
RequestID	ID заявки
ErrorCode	Код ошибки
ErrorMessage	Сообщение об ошибке
Code	Код ограничителя

*Описание структуры ответа на запрос удаления ограничителя по времени для карты*

	ТР _____	
--	----------	--

## 2.5 Операции

### 2.5.1 Получение списка операций по договору за период

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetOperByContract*

Адрес метода:

*/api/emv/v2/GetOperByContract?u={u}&contract={contract}&begin={begin}&end={end}&type={type}*

HTTP-метод	Обяз.	GET
contract	Да	Код договора
u	Да	Логин от ЛК
begin	Да	Дата начала периода в формате уууу-ММ-ддТНН:мм:сс. Период должен быть меньше 2х месяцев. Например, выборка данных за январь и февраль включительно: begin = 2017-01-01T00:00:00; end = 2017-02-01T23:59:59
end	Да	Дата окончания периода в формате уууу-ММ-ддТНН:мм:сс. Период должен быть меньше 2х месяцев. Например, выборка данных за январь и февраль включительно: begin = 2017-01-01T00:00:00; end = 2017-02-01T23:59:59
type	Нет	Тип возвращаемого результата XML или JSON.

Таблица 22. Описание входящих параметров запроса операций по договору за период

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;Operations&gt;   &lt;Oper Date="2016-07-01T20:50:17" Card="7826010000000436" Type="24" Value="3" Sum="29.70" DSum="29.70" Price="9.90" DPrice="9.90" Contract="ISS01TEST" Address="Россия, Московская область, г. Балашиха, Балашихинский р-н, ш.Энтузиастов, вл. 1, трасса М7, 18 км, Москва-Казань-Уфа ("Волга=""), справа" GCode="gkafe" GName="Кафе" Holder="" Code="9422" DTL="Молоко 50мл" Ref="9418" GCat="SERVICE" PosCode="AZS102328" CardGrp="Седельные тягачи" Vat="20.00"/&gt;   &lt;Oper Date="2016-07-01T20:50:13" Card="7826010000000436" Type="11" Value="3" Sum="29.70" DSum="29.70" Price="9.90" DPrice="9.90" Contract="ISS01TEST" Address="Россия, Московская область, г. Балашиха, Балашихинский р-н, ш.Энтузиастов, вл. 1, трасса М7, 18 км, Москва-Казань-Уфа ("Волга=""), справа" GCode="gkafe" GName="Кафе" Holder="" Code="9418" DTL="Молоко 50мл" Ref="" GCat="SERVICE" PosCode="AZS102328" CardGrp="Седельные тягачи" Vat="20.00"/&gt; &lt;/Operations&gt;</pre>
JSON	<pre>{   "OperationList": [     {       "Date": "2016-07-01T20:50:17",       "Card": "7826010000000436",       "Type": 24,       "Value": 3.0,       "Sum": 29.70,       "DSum": 1.50,       "Price": 9.90,       "DPrice": 0.50,       "Contract": "ISS01TEST",       "Address": "Россия, Московская область, г. Балашиха, Балашихинский р-н, ш.Энтузиастов, вл. 1, трасса М7, 18 км, Москва-Казань-Уфа (\\"Волга\\"), справа",       "GCode": "gkafe",       "GName": "Кафе",       "Holder": "",       "Code": "9422",       "DTL": "Молоко 50мл",       "Ref": "9418",</pre>

	2.0.0.32	47 / 65
--	----------	---------

	TP _____	
--	----------	--

	<pre> "GCat": "SERVICE", "PosCode": "AZS102328", "CardGrp": "Седельные тягачи", "Vat": 20.0 }, {   "Date": "2016-07-01T20:50:13",   "Card": "782601000000436",   "Type": 11,   "Value": 3.0,   "Sum": 29.70,   "DSum": 1.50,   "Price": 9.90,   "DPrice": 0.50,   "Contract": "ISS01TEST",   "Address": "Россия, Московская область, г. Балашиха, Балашихинский р-н, ш.Энтузиастов, вл. 1, трасса М7, 18 км, Москва-Казань-Уфа (\"Волга\"), справа",   "GCode": "gkafe",   "GName": "Кафе",   "Holder": "",   "Code": "9418",   "DTL": "Молоко 50мл",   "Ref": "",   "GCat": "SERVICE",   "PosCode": "AZS102328",   "CardGrp": "Седельные тягачи",   "Vat": 20.0 } ] </pre>
--	---

Описание полей	
Date	Дата операции (часовой пояс терминала)
Card	Идентификатор карты
Type	Тип операции (1 – Пополнение счета; 4 – списание со счета; 11- обслуживание, 24 – возврат на счет)
Value	Объем
Sum	Сумма
DSum	Размер скидки от суммы по цене АЗС. Отрицательное значение – наценка, положительное – скидка
Price	Цена
DPrice	Размер скидки от цены АЗС. Отрицательное значение – наценка, положительное - скидка
Contract	Код договора
Address	Адрес ТО
GCode	Код услуги
GName	Название услуги
Holder	Держатель карты
Code	Id операции
DTL	Расшифровка (детализация) услуги
Ref	Ссылка на главную операцию
GCat	Код категории товара
PosCode	Идентификатор АЗС
CardGrp	Список групп, в которых присутствует карта
Vat	НДС. Только для операций обслуживания (типы операций 11 и 24)

Таблица 23. Описание результата выполнения запроса карт договора

Источником транзакций является кэширующая БД ЛК, а не процессинг, следовательно, транзакции будут доступны с задержкой до 2х часов.

	2.0.0.32	48 / 65
--	----------	---------



	ТР _____	
--	----------	--

Целевая модель интеграции – это загрузка транзакций в собственное хранилище и уже его нужно использовать для построения внутренних отчётов и т.д.

## 2.5.2 Получение списка операций по договору по дате изменения

Получение списка операций по договору по дате импорта/изменения операции в системе. В выборку попадут операции, у которых дата импорта в систему (не дата операции) или дата изменения (корректировки, пересчёта цены) последние lastHours часов.

Источником транзакций является кэширующая БД ЛК, а не процессинг, следовательно, транзакции будут доступны с задержкой до 2х часов.

Целевая модель интеграции – это загрузка транзакций в собственное хранилище и уже его нужно использовать для построения внутренних отчётов и т.д.

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetOperByContractLM*

Адрес метода:

*/api/emv/v2/GetOperByContractLM?u={u}&contract={contract}&lastHours={lastHours}&type={type}*

HTTP-метод	Обяз.	GET
contract	Да	Код договора
u	Да	Логин от ЛК
lastHours	Да	Количество часов выборки. Первая дата округляется до целых часов. Например, если в 17:50 сформировать отчёт с lastHours = 4, в отчёт попадут операции, у которых дата импорта/изменения с 13:00 по 17:50. Максимальное значение: 120 часов.
type	Нет	Тип возвращаемого результата XML или JSON.

Таблица 243. Описание входящих параметров запроса операций по договору по дате изменения

### Возвращаемый результат:

Результат содержит структуру, аналогичную методу *GetOperByContract* [п.2.5.1](#)

## 2.6 Прочие

### 2.6.1 Получение списка услуг

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetGoodsList*

Адрес метода: */api/emv/v1/GetGoodsList?u={u}& type={type}*

HTTP-метод	Обяз.	GET
u	Да	Логин от ЛК
type	Нет	Тип возвращаемого результата XML или JSON.

Таблица 25. Описание входящих параметров метода получения списка услуг

### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;ArrayOfGoods xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;Goods Name="Мойка" Code="gmoika" Cat="SERVICE"/&gt;   &lt;Goods Name="АИ-98" Code="g98" Cat="FUEL"/&gt;   &lt;Goods Name="АИ-95" Code="g95" Cat="FUEL"/&gt;   &lt;Goods Name="АИ-92" Code="g92" Cat="FUEL"/&gt;</pre>

	TP _____	
--	----------	--

	<pre> &lt;Goods Name="АИ-80" Code="g80" Cat="FUEL"/&gt; &lt;Goods Name="ДТ" Code="gDT" Cat="FUEL"/&gt; &lt;Goods Name="СУГ" Code="gpropan" Cat="FUEL"/&gt; &lt;Goods Name="Шиномонтаж" Code="gshn" Cat="SERVICE"/&gt; &lt;Goods Name="Гостиница" Code="ggost" Cat="SERVICE"/&gt; &lt;Goods Name="Кафе" Code="gkafe" Cat="SERVICE"/&gt; &lt;Goods Name="Стоянка" Code="gstoyanka" Cat="SERVICE"/&gt; &lt;Goods Name="Пылесос" Code="gpylesos" Cat="SERVICE"/&gt; &lt;Goods Name="Прочие услуги" Code="gother" Cat="SERVICE"/&gt; &lt;Goods Name="Сопут Тов" Code="gsoput" Cat="GOODS"/&gt; &lt;Goods Name="АИ-95-Фирм" Code="g95f" Cat="FUEL"/&gt; &lt;Goods Name="АИ-92-Фирм" Code="g92f" Cat="FUEL"/&gt; &lt;Goods Name="ДТ-Зим" Code="gDTz" Cat="FUEL"/&gt; &lt;Goods Name="AdBlue" Code="gAdBlue" Cat="GOODS"/&gt; &lt;Goods Name="АИ-98-Фирм" Code="g98f" Cat="FUEL"/&gt; &lt;Goods Name="ДТ-Фирм" Code="gDTf" Cat="FUEL"/&gt; &lt;Goods Name="Оплата дорог" Code="gDorog" Cat="SERVICE"/&gt; &lt;Goods Name="ДТ-Аркт" Code="gDTa" Cat="FUEL"/&gt; &lt;Goods Name="ДТ-Меж" Code="gDTm" Cat="FUEL"/&gt; &lt;Goods Name="Масло" Code="gMasla" Cat="GOODS"/&gt; &lt;Goods Name="Метан" Code="gmetan" Cat="FUEL"/&gt; &lt;Goods Name="ТПБ" Code="gDT4" Cat="FUEL"/&gt; &lt;Goods Name="ТМСД" Code="gDTs4" Cat="FUEL"/&gt; &lt;Goods Name="Автоаксесуары" Code="gAvtoakses" Cat="GOODS"/&gt; &lt;/ArrayOfGoods&gt; </pre>
JSON	<pre> [   {     "Name": "Мойка",     "Code": "gmoika",     "Cat": "SERVICE"   },   {     "Name": "АИ-98",     "Code": "g98",     "Cat": "FUEL"   } ] </pre>
<b>Описание полей</b>	
Name	Наименование товара/услуги
Code	Код товара/услуги
Cat	Категория товара/услуги

Таблица 26. Описание входящих параметров метода получения списка услуг

## 2.6.2 Получение статуса заявки

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetStatusRequest*

Адрес метода: */api/emv/v1/GetStatusRequest?u={u}&id={id}&type={type}*

HTTP-метод	Обяз.	GET
u	Да	Логин от ЛК
Id	Да	Идентификатор заявки
type	Нет	Тип возвращаемого результата XML или JSON

Таблица 27. Описание входящих параметров метода получения статуса заявки

### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<RequestStatus>

	2.0.0.32	50 / 65
--	----------	---------

	TP _____	
--	----------	--

	<div>&lt;StatusCode&gt;P&lt;/StatusCode&gt; &lt;StatusName&gt;Заявка выполнена&lt;/StatusName&gt; &lt;/RequestStatus&gt;</div>	
JSON	{ <span>StatusCode</span> :"P", <span>StatusName</span> :"Заявка выполнена"}	
Описание полей		
StatusCode	<div>Код статуса. N - Новая заявка M - Ожидается ручная обработка, после неудачной E - Заявка с ошибочными/неполными данными P - Заявка выполнена S - Заявка остановлена C - Заявка в обработке U -Статус не определён A - Заявка на рассмотрении R - Ожидается настройка роуминга</div>	
StatusName	Статус	

Таблица 28. Описание результата метода получения статуса заявки

2.6.3 Получение кода договора по номеру карты

Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetContractByCard*

Адрес метода: */api/emv/v2/GetContractByCard?u={u}&card={card}&type={type}*

Параметр	Тип	Обяз.	GET
u	string	Да	Логин от ЛК
card	string	Да	Графический номер карты
type	string	Нет	Тип возвращаемого результата XML или JSON

Таблица 29. Описание входящих параметров метода получения кода договора по его карте

Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<String>ISS01TEST</String>
JSON	"ISS01TEST"
Описание полей	
Возвращается	Код договора

Таблица 30. Описание результата метода получения кода договора по его карте

Пример на 1С 8.3

```
Процедура КнопкаВыполнитьНажатие(Кнопка)
    Сервер = "lkapi.m-card.ru";
    Порт = Неопределено;
    Пользователь = Неопределено;
    Пароль = Неопределено;
    Прокси = Неопределено;
    Таймаут = 5;
    ЗащищенноеСоединение = Новый ЗащищенноеСоединениеOpenSSL();

    HTTPСоединение = Новый HTTPСоединение(Сервер, Порт, Пользователь, Пароль, Прокси, Таймаут, ЗащищенноеСоединение);

    Логин = "1111zzzzz";
    ПарольЛК = "zzzzzz";
    НомерКарты = "782601070000000006";

    Заголовки = Новый Соответствие;
    Заголовки.Вставить("RnCard-Identity-Account-Pass", СоздатьСтрокуBase64(ПарольЛК));

    // Get-запрос.

    HTTPЗапрос = Новый HTTPЗапрос("api/emv/v2/GetContractByCard?u=" + Логин + "&card=" + НомерКарты, Заголовки);
    HTTPОтвет = HTTPСоединение.Получить(HTTPЗапрос);

    КодСостояния = HTTPОтвет.КодСостояния;
    ТелоОтвета = HTTPОтвет.ПолучитьТелоКакСтроку(КодировкаТекста.UTF8);

    Если КодСостояния <> 200 Тогда
        Сообщить("Ошибка выполнения запроса. Код ответа: " + Строка(КодСостояния) + " Описание: " + ТелоОтвета);
    Иначе
        Сообщить(ТелоОтвета);
    КонецЕсли;

КонецПроцедуры

// Функция преобразует строку в Base64-строку.
//
// Параметры
// Строка - Строка - входная строка.
//
// Возвращаемое значение:
// Строка - строка, закодированная по алгоритму base64.
//
Функция СоздатьСтрокуBase64(Строка) Экспорт
    ВременныйФайл = ПолучитьИмяВременногоФайла();

    ЗаписьТекста = Новый ЗаписьТекста(ВременныйФайл, "CESU-8");
    ЗаписьТекста.Записать(Строка);
    ЗаписьТекста.Закрыть();

    ДвоичныеДанныеФайла = Новый ДвоичныеДанные(ВременныйФайл);
    СтрокаBase64 = Base64Строка(ДвоичныеДанныеФайла);

    УдалитьФайлы(ВременныйФайл);

    Возврат СтрокаBase64;

КонецФункции // СоздатьСтрокуBase64()
```

2.6.4 Получение информации о странах и их регионах

Метод и его входные параметры

HTTP-метод: *GET*  
Имя метода: *GetCountry*  
Адрес метода: */api/emv/v2/GetCountry?u={u}&contract={contract}&type={type}*

Параметр	Тип	Обяз.	GET
u	string	Да	Логин от ЛК
Contract	string	Да	Код договора
Type	string	Нет	Тип возвращаемого результата XML или JSON

Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;Country xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;CountryList&gt;     &lt;Item CountryCode="RUS" CountryName="Россия"&gt;       &lt;RegionList&gt;         &lt;Item RegionCode="26" RegionName="Республика Ингушетия" /&gt;         &lt;Item RegionCode="10" RegionName="Амурская область" /&gt;       &lt;/RegionList&gt;     &lt;/Item&gt;     &lt;Item CountryCode="UKR" CountryName="Украина"&gt;       &lt;RegionList&gt;         &lt;Item RegionCode="UA-30" RegionName="Киев" /&gt;         &lt;Item RegionCode="UA-05" RegionName="Винницкая область" /&gt;       &lt;/RegionList&gt;     &lt;/Item&gt;   &lt;/CountryList&gt; &lt;/Country&gt;</pre>
JSON	<pre>{   "CountryList": [     {       "CountryCode": "RUS",       "CountryName": "Россия",       "RegionList": [         {           "RegionCode": "26",           "RegionName": "Республика Ингушетия"         },         {           "RegionCode": "10",           "RegionName": "Амурская область"         }       ]     },     {       "CountryCode": "UKR",       "CountryName": "Украина",       "RegionList": [         {           "RegionCode": "UA-30",           "RegionName": "Киев"         },         {           "RegionCode": "UA-05",           "RegionName": "Винницкая область"         }       ]     }   ] }</pre>

	TP _____	
--	----------	--

	<pre>     }   ] } ] } </pre>
Описание полей	
CountryList	Список стран
CountryCode	Код страны
CountryName	Название страны
RegionList	Список(массив) регионов страны
RegionCode	Код региона
RegionName	Название региона

Таблица 31. Описание результата метода получения кода договора по его карте

## 2.6.5 Получение списка партнёров

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetPartners*

Адрес: */api/emv/v2/GetPartners?u={u}&contract={contract}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
type	Нет	Тип возвращаемого результата XML или JSON.

### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат	
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ArrayOfPartners xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;Partners ID="1234123" Code="765432_000001" Type="E" Name="Партнёрская сеть 66" PosURL="https://lk.rn-card.ru/Documents/interhost66.xlsx" /&gt;   &lt;Partners ID="" Code="H2H_GOODNEFT_EXCLUDE" Type="H" Name="Партнёрская сеть 55" PosURL="https://lk.rn-card.ru/Documents/interhost55.xlsx" /&gt; &lt;/ArrayOfPartners&gt;</pre>	
JSON	<pre>[   {     "ID": "1234123",     "Code": "765432_000001",     "Type": "E",     "Name": "Партнёрская сеть 66",     "PosURL": "https://lk.rn-card.ru/Documents/interhost66.xlsx"   },   {     "ID": "",     "Code": "H2H_GOODNEFT_EXCLUDE",     "Type": "H",     "Name": "Партнёрская сеть 55",     "PosURL": "https://lk.rn-card.ru/Documents/interhost55.xlsx"   } ]</pre>	
Описание полей		
Параметр	Тип	Описание
ID	string	ID партнёра. Только для партнёров с эквайринговыми данными
Code	string	Код партнёра

	ТР _____	
--	----------	--

Type	string	Н – межхостовой партнёр (без эквайринговых данных), Е – партнёр с эквайринговыми данными
Name	string	Наименование партнёра
PosURL	string	Ссылка на список ТО партнёра

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

Пример кода на PowerShell:

```
$result1 = $null
$uri = 'https://lkapi.rn-card.ru/api/emv/v2/GetPartners?u=ЛОГИН&contract=ISS000000&type=XML'
$headers = @{
    'RnCard-Identity-Account-Pass' = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes('ПАРОЛЬ'))
}
$result1 = Invoke-WebRequest -Uri $uri -Method Get -Headers $headers
$result1.Content
```

## 2.6.6 Получение списка НПО

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetNPOList*

Адрес: */api/emv/v2/GetNPOList?u={u}&contract={contract}&topContractCode={topContractCode}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
topContractCode	Нет	Возможные значения: 1. Код партнёра, работающего через эквайринговые данные (см. метод GetPartners <a href="#">п. 2.6.5</a> ) 2. RNC в остальных случаях.
type	Нет	Тип возвращаемого результата XML или JSON.

### Возвращаемый результат

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ArrayOfNPO xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;NPO ID="1481863" Code="100583_NP018" Name="000 Гуднефть-Юг" TopContractCode="100583_000001" /&gt;   &lt;NPO ID="466399" Code="NP022" Name="ЧТО Мойки Москвы" TopContractCode="RNC" /&gt;   &lt;NPO ID="466316" Code="NP019" Name="Алтайнефтепродукт" TopContractCode="RNC" /&gt; &lt;/ArrayOfNPO&gt;</pre>
JSON	<pre>[   {     "ID": "1481863",     "Code": "100583_NP018",     "Name": "000 Гуднефть-Юг",     "TopContractCode": "100583_000001"   },   {     "ID": "466399",     "Code": "NP022",     "Name": "ЧТО Мойки Москвы",     "TopContractCode": "RNC"   },   {     "ID": "466316",     "Code": "NP019",</pre>

	ТР _____	
--	----------	--

	<pre>"Name": "Алтайнефтепродукт", "TopContractCode": "RNC" } ]</pre>	
Описание полей		
Параметр	Тип	Описание
ID	string	ID НПО
Code	string	Код НПО
Name	string	Наименование партнёр
TopContractCode	string	Возможные значения: 1. Код партнёра, работающего через эквайринговые данные (см. метод GetPartners <a href="#">п. 2.6.5</a> ) 2. RNC в остальных случаях.

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

Пример кода на PowerShell:

```
$result1 = $null
$uri = 'https://lkapi.rn-card.ru/api/emv/v2/GetNPOList?u=ЛОГИН&contract=ISS00000&type=XML'
$headers = @{
    'RnCard-Identity-Account-Pass' = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes('ПАРОЛЬ'))
}
$result1 = Invoke-WebRequest -Uri $uri -Method Get -Headers $headers
$result1.Content
```

## 2.6.7 Получение списка ТО

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetPOSList*

Адрес: */api/emv/v2/GetPOSList?u={u}&contract={contract}&countryCode={countryCode}&regionCode={regionCode}&npoCode={npoCode}&posCode={posCode}&topContractCode={topContractCode}&flagActiveOnly={flagActiveOnly}&goodsInfo={goodsInfo}&type={type}*

Параметр	Обяз.	Значение	Примечание
u	Да	Логин от ЛК	
contract	Да	Код договора	
countryCode	Нет	Код страны	Рекомендуется использовать поля для фильтрации списка ТО.
regionCode	Нет	Код региона	
npoCode	Нет	Код НПО	
posCode	Нет	Код ТО	
topContractCode	Нет	Возможные значения: <ol style="list-style-type: none"> <li>Код партнёра, работающего через эквайринговые данные (см. метод GetPartners <a href="#">п. 2.6.5</a>)</li> <li>RNC в остальных случаях.</li> </ol>	
flagActiveOnly	Нет	Флаг: показывать только активные ТО. Возможные значения: <ol style="list-style-type: none"> <li>«Y» - показывать только активные ТО.</li> <li>«N» - показывать все ТО.</li> </ol> Значение по умолчанию: «Y».	
goodsInfo	Нет	Флаг (Y/N). Значение по умолчанию: «N». Показывать информацию по товарам/услугам: <ol style="list-style-type: none"> <li>Список доступных товаров/услуг</li> <li>Цены на товары/услуги</li> <li>Список дополнительных услуг</li> </ol>	
type	Нет	Тип возвращаемого результата XML или JSON.	

### Возвращаемый результат

	2.0.0.32	56 / 65
--	----------	---------



	ТР _____	
--	----------	--

В случае отсутствия сообщения об отказе, результат будет содержать следующую структуру:

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ArrayOfPOS xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;POS ID="470184" Code="AZS101953" Name="34" Type="АЗК" BrandName="РОСНЕФТЬ" CountryCode="RUS" RegionCode="42" Address="Россия, Липецкая область, Становлянский район, М4, 336 км, слева, Становлянский р- н, с. Бабарыкино" Address2="RUS, Липецкая область, BABARYKINO, Россия, Липецкая область, Становлянский р-он, с. Бабарыкино" RouteName="М4 &amp;quot;Дон&amp;quot; (Москва - Воронеж - Ростов-на-Дону - Краснодар - Джубга - Новороссийск)" RoutePosition="336" Latitude="52.952691" Longitude="38.237475" NPOCode="NPO000030" TopContractCode="RNC" FlagActive="Y" FlagNDS="Y"&gt;   &lt;GoodsList&gt;     &lt;Item&gt;g92&lt;/Item&gt;     &lt;Item&gt;g95&lt;/Item&gt;     &lt;Item&gt;gDT&lt;/Item&gt;   &lt;/GoodsList&gt;   &lt;ServiceList&gt;     &lt;Item&gt;Магазин&lt;/Item&gt;     &lt;Item&gt;Туалет&lt;/Item&gt;   &lt;/ServiceList&gt;   &lt;GoodsPriceList&gt;     &lt;Item GoodsCode="g95" Price="50.1" BeginDate="2021-10- 14T00:00:00" LoadDate="2021-10-24T03:58:39" /&gt;     &lt;Item GoodsCode="g92" Price="46.2" BeginDate="2021-10- 14T00:00:00" LoadDate="2021-10-24T03:58:39" /&gt;     &lt;Item GoodsCode="gDT" Price="48.5" BeginDate="2021-10- 14T00:00:00" LoadDate="2021-10-24T03:58:39" /&gt;   &lt;/GoodsPriceList&gt;   &lt;CardTypeList&gt;     &lt;Item&gt;Топливная карта Роснефть&lt;/Item&gt;     &lt;Item&gt;Виртуальная карта&lt;/Item&gt;   &lt;/CardTypeList&gt; &lt;/POS&gt; &lt;/ArrayOfPOS&gt;</pre>
JSON	<pre>[   {     "ID": "470184",     "Code": "AZS101953",     "Name": "34",     "Type": "АЗК",     "BrandName": "РОСНЕФТЬ",     "CountryCode": "RUS",     "RegionCode": "42",     "Address": "Россия, Липецкая область, Становлянский район, М4, 336 км, слева, Становлянский р-н, с. Бабарыкино",     "Address2": "RUS, Липецкая область, BABARYKINO, Россия, Липецкая область, Становлянский р-он, с. Бабарыкино",     "RouteName": "М4 \\"Дон\\" (Москва - Воронеж - Ростов-на-Дону - Краснодар - Джубга - Новороссийск)",     "RoutePosition": 336.0,     "Latitude": 52.952691,     "Longitude": 38.237475,     "NPOCode": "NPO000030",     "TopContractCode": "RNC",     "FlagActive": "Y",     "FlagNDS": "Y",     "GoodsList": [ "g92", "g95", "gDT" ],     "ServiceList": [ "Магазин", "Туалет" ],     "GoodsPriceList": [</pre>

	2.0.0.32	57 / 65
--	----------	---------

	TP _____	
--	----------	--

	<pre> {   "GoodsCode": "g95",   "Price": 50.1,   "BeginDate": "2021-10-14T00:00:00",   "LoadDate": "2021-10-24T03:58:39" }, {   "GoodsCode": "g92",   "Price": 46.2,   "BeginDate": "2021-10-14T00:00:00",   "LoadDate": "2021-10-24T03:58:39" }, {   "GoodsCode": "gDT",   "Price": 48.5,   "BeginDate": "2021-10-14T00:00:00",   "LoadDate": "2021-10-24T03:58:39" } ], "CardTypeList": [ "Топливная карта Роснефть", "Виртуальная карта" ] } ]</pre>
--	---

Описание полей		
Параметр	Тип	Описание
ID	string	ID ТО
Code	string	Код ТО
Name	string	Наименование ТО
Type	string	Тип ТО (АЗС/АЗК/Терминал услуг/ГАЗК и т.д.)
BrandName	string	Бренд ТО
CountryCode	string	Трёхбуквенный код страны (см. метод GetCountryList <a href="#">п. 2.6.4</a> )
RegionCode	string	Код региона (см. метод GetCountryList <a href="#">п. 2.6.4</a> )
Address	string	Адрес ТО
Address2	string	Адрес ТО в процессинговой системе.
RouteName	string	Наименование трассы
RoutePosition	decimal	Километр трассы
Latitude	decimal	Широта
Longitude	decimal	Долгота
NPOCode	string	Код НПО (см. метод GetNPOList <a href="#">п. 2.6.6</a> )
TopContractCode	string	Возможные значения: 1. Код партнёра, работающего через эквайринговые данные (см. метод GetPartners <a href="#">п. 2.6.7</a> ) 2. RNC в остальных случаях.
FlagActive	string	Флаг активности ТО: Y – активно, N – не активно.
FlagNDS	string	Флаг: АЗС работает с НДС
GoodsList	List<string>	Список доступных на ТО товаров/услуг (см. метод GetGoodsList <a href="#">п. 2.6.1</a> ). Отображаются только при значении параметра GoodsInfo = «Y»
ServiceList	List<string>	Список дополнительных услуг (мойка, магазин, кафе итд) Отображаются только при значении параметра GoodsInfo = «Y»
GoodsPriceList	List<GoodsPrice>	Список цен на товары/услуги на ТО Отображаются только при значении параметра GoodsInfo = «Y»
GoodsCode	string	Код товара/услуги (см. метод GetGoodsList <a href="#">п. 2.6.1</a> )
Price	decimal	Цена на товара/услуги
BeginDate	DateTime	Дата начала действия цены в формате YYYY-MM-DDTHH:mm:ss.

	2.0.0.32	58 / 65
--	----------	---------

	ТР _____	
--	----------	--

LoadDate	DateTime	Дата последней актуализации цены в формате YYYY-MM-DDTHH:mm:ss.
CardTypeList	List<string>	Разрешенные типы карт (топливные карты Роснефть, виртуальные карты, онлайн-оплата из мобильного приложения РН-Карт, пластиковая карта в мобильном приложении РН-Карт итд)

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

Пример кода на PowerShell получения информации по ТО, принадлежащее партнёру с кодом 123123\_987654 в Новосибирской области:

```
$result1 = $null
$uri = 'https://lkapi.rn-card.ru/api/emv/v2/GetPosList?u=ЛОГИН&contract=ISS00000&countryCode=RUS&regionCode=50&topContractCode=123123_987654&goodsInfo=y&type=JSON'
$headers = @{
    'RnCard-Identity-Account-Pass' = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes('ПАРОЛЬ'))
}
$result1 = Invoke-WebRequest -Uri $uri -Method Get -Headers $headers
$result1.Content
```

## 2.6.8 Получение информации о лимитах и/или ограничителях всех карт договора, одной карты и/или договора

### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *GetAllLimiters*

Адрес метода:

*/api/emv/v2/GetAllLimiters?u={u}&contract={contract}&card={card}&flagLimits={flagLimits}&flagGoodsRestrictions={flagGoodsRestrictions}&flagRegionRestrictions={flagRegionRestrictions}&flagContract={flagContract}&flagCard={flagCard}&type={type}*

Параметр	Обяз.	Значение
u	Да	Логин от ЛК
contract	Да	Код договора
card	Нет	Номер карты
flagLimits	Нет	Флаг: Y – возвращать лимиты (количественные ограничители), N – не возвращать. По умолчанию Y
flagGoodsRestrictions	Нет	Флаг: Y – возвращать товарные ограничители (разрешения), N – не возвращать. По умолчанию Y
flagRegionRestrictions	Нет	Флаг: Y – возвращать региональные ограничители (разрешения), N – не возвращать. По умолчанию Y
flagContract	Нет	Флаг: Y – возвращать запрошенные данные по договору, N – не возвращать. По умолчанию Y
flagCard	Нет	Флаг: Y – возвращать запрошенные данные карте/картам, N – не возвращать. По умолчанию Y
type	Нет	Тип возвращаемого результата XML или JSON. Значения: xml; json; (Необязательный параметр)

Карта или договор не будут возвращены в списке, если по ним нет лимитов и ограничителей.

1. При FlagContract = Y, FlagCard = Y, без Card будут возвращены лимиты и ограничители по договору и по всем картам.
2. При FlagContract = N, FlagCard = Y, без Card будут возвращены лимиты и ограничители по всем картам.
3. При FlagContract = Y, FlagCard = Y, с Card будут возвращены лимиты и ограничители по договору и по карте Card.

	2.0.0.32	59 / 65
--	----------	---------

	TP _____	
--	----------	--

4. При FlagContract = N, FlagCard = Y, с Card будут возвращены лимиты и ограничители по карте Card.

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;ArrayOfLimiters xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"&gt;   &lt;Limiters Contract="ISS123456"&gt;     &lt;Limits&gt;       &lt;Item Code="TEMPL_RLI_ITEM4" Currency="V" GCat="gDTa,gDT,gDTz,gDTm" GFlag="G" Prd="N" Val="1396" CurValue="568.02" MaxNumber="0" CurNumber="0" FlagRead="N" /&gt;       &lt;Item Code="TEMPL_RLI_ITEM5" Currency="V" GCat="g92" GFlag="G" Prd="N" Val="1395" CurValue="829.38" MaxNumber="0" CurNumber="0" FlagRead="N" /&gt;     &lt;/Limits&gt;     &lt;GoodsRestrictions&gt;       &lt;Item GCat="gDT,gDTa,gDTz,gDTm" GFlag="G" Id="1098234" Type="E" FlagRead="N" /&gt;       &lt;Item GCat="g92" GFlag="G" Id="1267123" Type="E" FlagRead="N" /&gt;     &lt;/GoodsRestrictions&gt;     &lt;RegionRestriction&gt;       &lt;Item CountryCode="" RegionCode="" NpoCode="NP0000068,NP0000070" PosCode="" RestrictionType="D" RestrictionID="1098234" FlagRead="N" PartnerCode="" /&gt;       &lt;Item CountryCode="RUS" RegionCode="28" NpoCode="" PosCode="" RestrictionType="E" RestrictionID="1098123" FlagRead="N" PartnerCode="" /&gt;     &lt;/RegionRestriction&gt;   &lt;/Limiters&gt; &lt;/Limiters Card="7826010123456789"&gt;   &lt;GoodsRestrictions&gt;     &lt;Item GCat="gAdBlue" GFlag="G" Id="2226123" Type="E" FlagRead="N" /&gt;   &lt;/GoodsRestrictions&gt; &lt;/Limiters&gt; &lt;/ArrayOfLimiters&gt;</pre>
JSON	<pre>[ {   "Contract": "ISS123456",   "Limits": [     {       "Code": "TEMPL_RLI_ITEM4",       "Currency": "V",       "GCat": "gDTa,gDT,gDTz,gDTm",       "GFlag": "G",       "Prd": "N",       "Val": 1396.0,       "CurValue": 568.02,       "MaxNumber": 0,       "CurNumber": 0,       "FlagRead": "N"     },     {       "Code": "TEMPL_RLI_ITEM5",       "Currency": "V",       "GCat": "g92",       "GFlag": "G",       "Prd": "N",       "Val": 1395.0,       "CurValue": 829.38,       "MaxNumber": 0,       "CurNumber": 0,       "FlagRead": "N"     }   ],   "GoodsRestrictions": [     {       "GCat": "gDT,gDTa,gDTz,gDTm",       "GFlag": "G",       "Id": "1098123",       "Type": "E",       "FlagRead": "N"     },     {       "GCat": "g92",</pre>

	<pre>         "GFlag": "G",         "Id": "1267123",         "Type": "E",         "FlagRead": "N"       }     ],     "RegionRestrictions": [       {         "CountryCode": "",         "RegionCode": "",         "NpoCode": "NPO000068,NPO000070",         "PosCode": "",         "RestrictionType": "D",         "RestrictionID": "1098123",         "FlagRead": "N",         "PartnerCode": ""       },       {         "CountryCode": "RUS",         "RegionCode": "28",         "NpoCode": "",         "PosCode": "",         "RestrictionType": "E",         "RestrictionID": "1098234",         "FlagRead": "N",         "PartnerCode": ""       }     ]   },   {     "Card": "7826010123456789",     "Limits": null,     "GoodsRestrictions": [       {         "GCat": "gAdBlue",         "GFlag": "G",         "Id": "2226123",         "Type": "E",         "FlagRead": "N"       }     ],     "RegionRestrictions": null   } ] </pre>
--	---

Описание полей			
Параметр	Тип	Описание	Примечание
Contract	string	Номер договора.	Заполнен Contract – информация по лимитам и/или ограничителям по договору.
Card	string	Номер карты	Заполнен Card – информация по лимитам и/или ограничителям по карте.
Limits		Список (массив) лимитов	см. метод GetCardLimits п. <a href="#">2.4.2.1</a>
GoodsRestrictions		Список (массив) товарных ограничителей	см. метод GetGoodsCardRestrictions п. <a href="#">2.4.3.1</a>
RegionRestrictions		Список (массив) региональных ограничителей	см. метод GetCardRegionRestriction п. <a href="#">2.4.9.1</a>
TimeRestrictions		Список (массив) ограничителей по времени	см. метод GetCardTimeRestriction п. <a href="#">2.4.10.1</a>

Описание структуры ответа на запрос получения списка лимитов и ограничителей договора и/или карт

В случае наличия сообщения об отказе, формируется сообщение с кодом ошибки HTTP-сервера. Описание ошибок приведено в п. 2.2 «Описание ошибок»

Пример кода на PowerShell получения лимитов и ограничителей по договору и по всем картам договора:

```

$result1 = $null
$uri = 'https://lkapi.rn-card.ru/api/emv/v1/GetAllLimiters?u=ЛОГИН&contract=ISS000000'
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$headers = @{
  'RnCard-Identity-Account-Pass' = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes('ПАРОЛЬ'))
}
$result1 = Invoke-WebRequest -Uri $uri -Method Get -Headers $headers
$result1.Content

```

	TP _____	
--	----------	--

## 2.7 Группы карт

### 2.7.1 Получение списка групп карт

#### Метод и его входные параметры

HTTP-метод: *GET*

Имя метода: *InfoCardGroup*

Адрес метода:

*/api/emv/v2/InfoCardGroup/{id}?u={u}&contract={contract}&showCards{showCards}&type={type}*

HTTP-метод	Обяз.	GET
id	Нет	ID группы карт. Если не указан, то результат будет содержать все группы карт по договору
u	Да	Логин от ЛК
contract	Да	Код договора
showCards	Нет	Отображать карты содержащиеся в группе(ах) Значение по умолчанию – N. Для отображения карт не входящих ни в одну группу, указать gid=0 и showCards=Y
type	Нет	Тип возвращаемого результата XML или JSON.

Таблица 32. Описание входящих параметров метода получения списка групп карт

#### Возвращаемый результат:

Результат содержит следующую структуру

Формат	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;Groups xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"&gt;   &lt;Group ID="1003294" Name="петя" Cnt="1"&gt;     &lt;Card ID="7826010700000000001" Status="Утеряна" Owner="ЭК" Type="A" LastOperDate="2017-06-08T09:40:24" /&gt;   &lt;/Group&gt; &lt;/Groups&gt;</pre>
JSON	<pre>{   "CardGroup": [     {       "ID": 1003294,       "Name": "петя",       "Cnt": 1,       "Cards": [         {           "ID": "7826010700000000001",           "Status": "Утеряна",           "Owner": "ЭК",           "LastOperDate": "2017-06-08T09:40:24"         }       ]     }   ] }</pre>
Описание полей	
ID	ID группы карт.
Name	Наименование группы карт
Cnt	Количество карт
Cards	Объект. Список карт в группе.
ID	Номер карты
Status	Статус карты
Owner	Примечание к карте
LastOperDate	Дата последней операции по карте. "0001-01-01T00:00:00" – не было операций

	2.0.0.32	62 / 65
--	----------	---------

	ТР _____	
--	----------	--

Таблица 33. Описание полей возвращаемых методом получения списка групп карт

## 2.7.2 Управление группами карт

Метод по входящим параметрам определяет возможность создания, переименования или удаления группы карт.

### Метод и его входные параметры

HTTP-метод	POST		
Имя метода	RequestGroupCard		
Адрес метода	https://lkapi.rn-card.ru/api/emv/v2/RequestGroupCard		
Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин от ЛК
Contract	string	Да	Код договора
GroupID	decimal	Усл.	ID группы карт. (Необязательное поле для операции создания группы)
GroupName	string	Усл.	Название группы карт. (Необязательное поле для операции удаления группы)
Action	char	Да	Операции с группой карт.

Таблица 34. Описание входящих параметров метода управления группами карт

### Возможные операции с группами карт

Код	Описание
C	Создание группы карт
R	Изменение наименования группы карт. Параметр GroupName должен содержать новое наименование группы
D	Удаление группы карт. (Параметр GroupName необязателен). Удаляются только группы, за которыми не закреплены карты

### Проверяемые ограничения на стороне ЛК для определения возможности создания заявки:

- 1) Наличие учётной записи.
- 2) Наличие оплаченной услуги «Информационное обслуживание» (Если предусмотрено договором)
- 3) В случае изменения наименования или удаления группы, проверяется принадлежность группы к договору пользователя в ЛК.
- 4) При удалении группы проверяется наличие карт, закрепленных за группой. Если группа содержит карту(ы), то удаление группы невозможно.

### Возвращаемый результат:

Результат будет содержать следующую структуру:

Формат результата	Результат
XML	<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;GroupCardStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" GroupID="1003322" /&gt;</pre>
JSON	<pre>{   "GroupID": 1003322 }</pre>

### Примеры запросов(CURL):

Создание группы карт	
Запрос	<pre>curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАП0ЛЬ_BASE64' -X POST -d 'U=demo &amp;Contract=ISS01TEST&amp;GroupName=testGroup2&amp;Action=C' http://lkapi.rn-card.ru/api/emv/v2/RequestGroupCard</pre>
Ответ	<pre>{   "GroupID":1012273.0 }</pre>
Изменение наименования группы карт	

	TP _____	
--	----------	--

<b>Запрос</b>	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&GroupID=1012273&GroupName=MyRenamedGroup&Action=R' http://lkapi.rn-card.ru/api/emv/v2/RequestGroupCard
<b>Ответ</b>	{ "GroupID": 1012273 }
<b>Удаление группы карт</b>	
<b>Запрос</b>	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&GroupID=1012268&Action=D' http://lkapi.rn-card.ru/api/emv/v2/RequestGroupCard
<b>Ответ</b>	{ "GroupID": -1.0 }

### 2.7.3 Управление картами в группах

Метод по входящим параметрам определяет возможность добавления, переноса или удаления карт из группы.

#### Метод и его входные параметры

HTTP-метод	POST		
Имя метода	RequestCardGroup		
Адрес метода	https://lkapi.rn-card.ru/api/emv/v2/RequestCardGroup		
Параметры метода			
Параметр	Тип	Обяз.	Описание
U	string	Да	Логин от ЛК
Contract	string	Да	Код договора
CardList[]	array	Да	Массив карт. Массив содержит графические номера карт.
GroupID	decimal	Да	ID группы карт
Action	char	Да	Вид операции с группой карт
TransferGroupID	decimal	Усл. обяз.	ID группы карт, в которую будут перемещаться карты. Параметр является обязательным только для операции переноса карт.
FlagRemoveOther	char	Нет	Возможные значения: Y, N. Определяет необходимость удаления карт из всех групп, в которых находится перемещаемая карта.

#### Возможные операции с группами карт

<b>Код</b>	<b>Описание</b>
A	Добавление карт в группу.
T	Перенос карт из одной группы в другую. Параметр TransferGroupID обязателен при данном типе операции.
D	Удаление карт из группы

#### Проверяемые ограничения на стороне ЛК для определения возможности создания заявки:

- Наличие учётной записи.
- Наличие оплаченной услуги «Информационное обслуживание». Если предусмотрено договором.

#### Возвращаемый результат:

Результат будет содержать следующую структуру:

<b>Формат результата</b>	<b>Результат</b>
XML	<?xml version="1.0" encoding="utf-8"?> <GroupCardStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" GroupID="1003322" />
JSON	{ "GroupID": 1003322 }

#### Примеры запросов(CURL):

	2.0.0.32	64 / 65
--	----------	---------



	TP _____	
--	----------	--

<b>Добавление карты в группу</b>	
<b>Запрос</b>	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&CardList[0]=7826010000000279&GroupID=1006580&Action=A' https://lkapi.rn-card.ru/api/emv/v2/RequestCardGroup
<b>Ответ</b>	{ "GroupID": 1006580 }
<b>Перенос карты из группы в группу</b>	
<b>Запрос</b>	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&CardList[0]=7826010000000279&GroupID=1006580&Action=T&TransferGroupID=1012268' https://lkapi.rn-card.ru/api/emv/v2/RequestCardGroup
<b>Ответ</b>	{ "GroupID": 1012268 }
<b>Удаление карты из группы</b>	
<b>Запрос</b>	curl -H 'Content-Type: application/x-www-form-urlencoded' -H 'RnCard-Identity-Account-Pass: ПАОЛЬ_BASE64' -X POST -d 'U=demo&Contract=ISS01TEST&CardList[0]=7826010000000279&GroupID=1006580&Action=D' https://lkapi.rn-card.ru/api/emv/v2/RequestCardGroup
<b>Ответ</b>	{ "GroupID": 1012268 }