

Система приема платежей Cyber

## Спецификация протокола XML API

Версия протокола 1.0

Версия документа: 2009.01.05.1

## Оглавление

1. Введение .....	3
2. Общие сведения о формате сообщений .....	3
2.1. Формат запросов клиентов.....	3
2.2. Формат ответов на запросы клиентов.....	4
3. Описание поддерживаемых вызовов .....	7
3.1. Загрузка справочников.....	7
3.2. Отправка телеметрии.....	8
3.3. Работа с платежами.....	9
3.4. Регистрация контрактов.....	12
3.5. Получение отчетов.....	12
3.6. Служебные вызовы.....	16

# 1 Введение

В данном документе полное описание протокола взаимодействия клиентов с сервером системы приема платежей

## 2 Общие сведения о формате сообщений

### 2.1 Формат запросов клиентов

Запрос клиента к шлюзу CyberXML API имеет вид:

```
<message>
  <head>
    <p name="username">USERNAME</p>
    <p name="password">PASSWORD</p>
    <p name="client_software">CLIENT_SOFTWARE</p>
    <p name="api_version">API_VERSION</p>
    <!--произвольное количество параметров заголовка-->
  </head>
  <body>
    <!--Произвольное количество блоков вызовов-->
  </body>
</message>
```

<i>Название параметра</i>	<i>Значение параметра</i>
USERNAME	имя пользователя
PASSWORD	пароль пользователя
CLIENT_SOFTWARE	версия и тип клиентского программного обеспечения
API_VERSION	версия реализации протокола клиента

Таблица 1: Значение обязательных атрибутов заголовка (head) сообщения

Каждое сообщение сообщение состоит из заголовка и собственно тела. Заголовок должен содержать ряд обязательных параметров, список которых приведен в табл. 1.

Тело сообщения может содержать произвольное количество вызовов процедур на стороне сервера. Каждый вызов должен быть оформлен следующим образом:

```
<call id="CALL_IDENTIFIER" name="CALL_NAME">
  <p name="PARAMETER_1_NAME">PARAMETER_1_VALUE</p>
  <p name="PARAMETER_2_NAME">PARAMETER_2_VALUE</p>
  <!--любое количество параметров на вход процедуры-->
</call>
```

<i>Название параметра</i>	<i>Значение параметра</i>
CALL_IDENTIFIER	уникальный идентификатор вызова в контексте сессии клиентского приложения
CALL_NAME	имя вызова — должно быть одним из разрешенных имен, список см. ниже
PARAMETER_1_NAME	имя параметра вызова #1
PARAMETER_1_VALUE	значение параметра вызова #1

Таблица 2: Параметры вызова процедуры

## 2.2 Формат ответов на запросы клиентов

Как и запрос клиента, ответ шлюза XML API состоит из заголовка сообщения и его тела.

Общий вид ответа шлюза:

```
<message>
  <head>
    <p name="code">RESULT_CODE</p>
    <p name="time">SERVER_TIME</p>
    <p name="api_version">API_VERSION</p>
    <p name="gateway">GATEWAY</p>
    <!--произвольное количество параметров заголовка-->
  </head>
  <body>
    <!--результаты выполненных вызовов-->
  </body>
</message>
```

Как и в случае запроса клиента, заголовок ответа сервера содержит несколько обязательных атрибутов и произвольное количество необязательных параметров.

<i>Название параметра</i>	<i>Значение параметра</i>
RESULT_CODE	код результата обработки запроса
SERVER_TIME	дата и время обработки запроса на сервере
API_VERSION	версия реализуемого шлюзом протокола
GATEWAY	URL рабочего шлюза XML API

Таблица 3: Обязательные атрибуты заголовка ответа шлюза XML API

*Клиентские реализации протокола должны отправлять сообщения на URL, указанный в параметре GATEWAY. В противном случае возможна потеря связи клиента с шлюзом XML API из-за изменения URL последнего.*

Значения кодов результата обработки сообщения приведены (значение параметра RESULT\_CODE) в табл. 4.

<i>Код</i>	<i>Значение</i>
800	запрос успешно обработан
801	неверный формат запроса клиента
802	доступ закрыт: неверный сертификат безопасности
803	доступ закрыт: неверно имя пользователя или пароль
804	указанный тип ПО не поддерживается
805	сервис временно недоступен, повторите запрос позже
806	указанная версия протокола не поддерживается
807	неверный формат сообщения
999	другая ошибка

Таблица 4: Значения кодов результата обработки сообщения

В случае успешной обработки запроса шлюзом, клиент получит непустое тело сообщения, состоящее из блоков вызовов процедур с выходными параметрами. Количество блоков выходных параметров в ответе равно количеству блоков входных параметров в запросе. Блок выходных параметров имеет вид:

```
<call id="CALL_IDENTIFIER" name="CALL_NAME" code="RESULT_CODE"
comment="COMMENT">
  <p name="PARAMETER_1_NAME">PARAMETER_1_VALUE</p>
  <p name="PARAMETER_2_NAME">PARAMETER_3_VALUE</p>
  <!-- прочие выходные параметры -->
</call>
```

<i>Название параметра</i>	<i>Значение параметра</i>
CALL_IDENTIFIER	уникальный идентификатор вызова в контексте сессии клиентского приложения
CALL_NAME	имя вызова
RESULT_CODE	код результата выполнения вызова
COMMENT	комментарий к результату выполнения вызова
PARAMETER_1_NAME	имя выходного параметра #1
PARAMETER_1_VALUE	значение выходного параметра #1

Таблица 5: Значения атрибутов блока выходных параметров

*Сопоставление блоков выходных параметров с конкретными вызовами клиентская реализация должна осуществлять по уникальным идентификаторам вызовов, которые совпадает в запросе и ответе*

Для всех вызовов действительны общие коды результатов обработки, приведенные в табл. 5. Коды, специфичные для каждого вызова приведены далее, при описании конкретных вызовов. Код "0" означает общий успех, конкретный результат зависит от контекста вызова.

Код "999" служит признаком прочих ошибок для процедур, суть ошибки может быть

приведена в поле COMMENT.

<i>Код</i>	<i>Значение</i>
0	вызов успешно выполнен
901	не найден обработчик для вызова
902	недостаточно привилегий для вызова
903	недостаточно параметров для вызова
904	неверны параметры вызова
905	неизвестная ошибка вызываемой процедуры
999	прочие ошибки

### 3 Описание поддерживаемых вызовов

<i>Имя вызова</i>	<i>Краткое описание</i>
ref_download	загрузка справочника
params_register	регистрация параметров клиента (мгновенная телеметрия)
srv_register_event	регистрация события, произошедшего на клиенте
pay_check	предварительная проверка платежа
pay_confirm	подтверждение (регистрация) ранее проверенного платежа
pay_verify	проверка статуса ранее зарегистрированного платежа
contract_register	регистрация контракта
report_pays	отчет "Поиск платежей"
report_recent_pays	отчет "Последние платежи"
report_simcards	отчет "Симкарты"
report_terminals	отчет "Терминалы"
report_dealer	отчет "Дилерский"
report_dealer_extended	отчет "Дилерский расширенный"
report_dealer_account	отчет "Корректировки счета дилера"
report_bank	отчет "Банковский"
srv_ping	

Таблица 6: Доступные вызовы XML API

Все вызовы можно разделить на 5 групп: загрузка справочников, отправка телеметрии, работа с платежами, регистрация контрактов, получение отчетов и служебные. Ниже рассмотрена подробно каждая из групп.

#### 3.1 Загрузка справочников

Единственный вызов служит для загрузки указанного в параметре справочника. Описание доступных справочников и их форматов выходит за рамки данного документа.

##### 3.1.1 ref\_download

<b>Описание</b>	<i>Вызов для получение справочника с именем, указанным в качестве входного параметра.</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	ref_name	имя запрашиваемого справочника
<i>Выходные</i>	ref_cdata	секция <![CDATA[ . . . ]]> с телом справочника
<b>Собственные коды результата</b>		
<i>Код</i>	<i>Значение</i>	
201	запрошенный справочник не найден	

Таблица 7: ref\_download – описание вызова

### 3.2 Отправка телеметрии

Вызовы группы служат для регистрации данных клиентского ПО на сервере процессингового центра.

#### 3.2.1 params\_register

<b>Описание</b>	<i>Вызов для регистрации клиентом произвольных данных на сервере процессингового центра. Регистрируемые данные могут иметь произвольную форму.</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	data_cdata	загружаемые параметры
<i>Выходные</i>	-	

Таблица 8: params\_register - описание вызова

#### 3.2.2 srv\_register\_event

<b>Описание</b>	<i>Вызов для получение справочника с именем, указанным в качестве входного параметра.</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	type	тип регистрируемого события
	date	дата события
	message	описание события
<i>Выходные</i>	-	

Таблица 9: *srv\_register\_event* - описание вызова

### 3.3 Работа с платежами

#### 3.3.1 pay\_check

Тип платежа представляет собой строку, значения всех принимаемых системой типов платежей могут быть получены из соответствующего справочника.

Способ обработки платежа может принимать два значения: 'ONLINE' или 'OFFLINE'. Обработка в режиме 'ONLINE' подразумевает получение однозначного ответа от провайдера (организации, обслуживающей абонента), возможен ли прием платежа с указанными входными параметрами. Обработка в режиме 'OFFLINE' не предполагает подобных проверок. Платеж будет буферизован системой для дальнейшей обработки и, по возможности, проведения. Фактически, в реальных условиях при плохом качестве связи из соображений эффективности рекомендуется применять способ обработки 'OFFLINE' везде, где это возможно.

Значение подтверждения повторного платежа может принимать значение 'TRUE' или 'FALSE'. Если в течение недавнего времени платеж на данный номер уже регистрировался в системе, то повторный платеж на тот же номер не будет проверен до тех пор, пока значение данного флага не будет установлено в 'TRUE'. В случае автоматического приема платежей (например, терминал самообслуживания) при регистрации в два этапа (проверка, затем подтверждение) данный флаг может быть установлен в значение 'TRUE' по умолчанию.

Поле параметров платежа должно содержать данные вида:

```
<fields>
  <f name="p1_name">p1_value</f>
</fields>
```

Данные должны быть заключены в секцию CDATA.

<b>Описание</b>	<i>Вызов осуществляет проверку нового платежа</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	pay_type	тип платежа
	params_cdata	параметры платежа
	processing_type	способ обработки платежа
	confirmation	флаг подтверждения платежа
<i>Выходные</i>	provider_cdata	реквизиты провайдера
	pay_id	идентификатор платежа
<b>Собственные коды результата</b>		
<i>Код</i>	<i>Значение</i>	
101	ошибка проверки платежа в режиме онлайн	
102	повторный платеж на тот же номер, требуется подтверждение	

Таблица 10: *pay\_check* - описание вызова

Реквизиты платежа в выходных параметрах имеют смысл только в случае обработки платежа в режиме 'ONLINE'. Полученный идентификатор платежа должен быть использован при подтверждении.

### 3.3.2 pay\_confirm

<b>Описание</b>	<i>Вызов регистрирует ранее проверенный платеж</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	pay_sum	принятая от клиента сумма платежа
	enter_sum	вносимая на счет сумма платежа
	processing_type	способ обработки платежа
	pay_id	номер платежа, полученный при проверке
<i>Выходные</i>	provider_cdata	реквизиты провайдера
<b>Собственные коды результата</b>		
<i>Код</i>	<i>Значение</i>	
113	ошибка регистрации платежа в режиме онлайн	

Таблица 11: pay\_confirm – описание вызова

При подтверждении платежа следует обратить внимание на вносимую провайдеру сумму (параметр enter\_sum). Для большинства платежей в данном поле должно быть указано значение 'NULL'. Это означает, что расчет суммы для внесения провайдеру будет произведен системой автоматически в зависимости от комиссии дилера для данного типа платежей.

Однако, существуют платежи, в которых сумма для внесения провайдеру фиксирована (например, платежи за услуги междугородной связи). В этом случае значение этой суммы должно быть указано в параметре enter\_sum.

### 3.3.3 pay\_verify

<b>Описание</b>	<i>Вызов для проверки состояния ранее зарегистрированного платежа</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
	pay_id	номер платежа, полученный при проверке
<i>Выходные</i>	reg_date	дата регистрации
	status	статус платежа
	status_date	дата статуса
	pay_sum	сумма платежа
	enter_sum	сумма к зачислению на счет
	comments	комментарий
	param_cdata	параметры платежа

Таблица 12: pay\_verify – описание вызова

### 3.4 Регистрация контрактов

#### 3.4.1 contract\_register

<b>Описание</b>	<i>Регистрирует контракт оператора связи в системе</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	sim_type	тип симкарты
	sim_number	номер симкарты
	manager_name	имя менеджера
	contract_cdata	данные контракта
<i>Выходные</i>	-	

Таблица 13: contract\_register – описание вызова

### 3.5 Получение отчетов

Клиентом могут быть запрошены несколько видов отчетов. Для этого используются вызовы данной группы.

Значение выходного параметра header\_cdata имеет вид:

```
<data>
<r>
```

```
<d>HEADER_DATA_00</d>
<d>HEADER_DATA_01</d>
</r>
</data>
```

Значение выходного параметра report\_cdata имеет вид:

```
<data>
<r>
<d>DATA_00</d>
<d>DATA_01</d>
</r>
<r>
<d>DATA_10</d>
<d>DATA_11</d>
</r>
<r>
<d>DATA_20</d>
<d>DATA_21</d>
</r>
</data>
```

Дата предоставляется в формате: DD.MM.YY

### 3.5.1 report\_pays

<b>Описание</b>	<i>Детальный отчет по платежам за указанный период</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	start_date	дата начала периода. Формат: DD.MM.YY
	end_date	дата конца периода. Формат: аналогично start_date
	payment_id	идентификатор платежа
	account_number	номер счета/телефона
	status	статус платежа
	point_name	имя пункта приема платежей
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 14: report\_pays – описание вызова

### 3.5.2 report\_recent\_pays

<b>Описание</b>	<i>Получить данные об указанном количестве последних платежей для данной точки приема</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	n	количество платежей, по которым требуются данные
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 15: report\_recent\_pays - описание вызова

### 3.5.3 report\_simcards

<b>Описание</b>	<i>Отчет по симкартам</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	start_date	дата начала периода
	end_date	дата конца периода
	simcard_number	номер симкарты
	point_name	имя точки
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 16: report\_simcards - описание вызова

### 3.5.4 report\_terminals

<b>Описание</b>	<i>Отчет с детальной информацией об аппаратах самообслуживания</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	-	
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 17: report\_terminals - описание вызова

### 3.5.5 report\_dealer

<b>Описание</b>	<i>Финансовый отчет дилера</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	start_date	дата начала периода
	end_date	дата конца периода
	point_name	имя точки приема платежей
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 18: report\_dealer - описание вызова

### 3.5.6 report\_dealer\_extended

<b>Описание</b>	<i>Расширенный отчет дилера</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	-	
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 19: report\_dealer\_extended

### 3.5.7 report\_dealer\_account

<b>Описание</b>	<i>Отчет по корректировкам состояния счета дилера</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	start_date	дата начала периода
	end_date	дата конца периода
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 20: report\_dealer\_account - описание вызова

### 3.5.8 report\_bank

<b>Описание</b>	<i>Отчет "Банк"</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	-	
<i>Выходные</i>	header_cdata	заголовок отчета
	report_cdata	данные отчета

Таблица 21: report\_bank - описание вызова

## 3.6 Служебные вызовы

### 3.6.1 srv\_ping

<b>Описание</b>	<i>Вызов для проверки доступности шлюза XML API</i>	
<b>Параметры</b>	<i>Имя</i>	<i>Значение</i>
<i>Входные</i>	-	
<i>Выходные</i>	-	

Таблица 22: srv\_ping - описание вызова