

ОАО «КИБЕРПЛАТ»

123610, г.Москва, ЦМТ-2,
Краснопресненская наб., д.12 подъезд № 7
Телефон: (495) 967-02-20
Факс: (495) 967-02-08



JSC CYBERPLAT

123610, Moscow, WTC-2,
Krasnopresnenskaya nab., 12, entrance # 7
Phone: (495) 967-02-20
Fax: (495) 967-02-08

**Руководство по программному взаимодействию с системой
«CyberPlat»**

Техническое описание протокола взаимодействия

Версия 1.6.5

В настоящем документе представлен протокол взаимодействия с платёжной системой CyberPlat® при совершении платежа и при проверке состояния платежа.

Содержание

1. Общие сведения о протоколе обмена.....	3
2. Формат сообщений.....	3
2.1. Структура сообщения.....	3
2.2. Получение разрешения на платеж (запрос на проверку номера телефона).....	4
2.3. Запрос на оплату.....	6
2.4. Проверка состояния платежа (запрос статуса).....	8
2.5. Значения поля «ERROR» (коды ошибок сервера Киберплат).....	10
3. Алгоритм работы программного обеспечения.....	11
4. Рекомендации платёжным агентам, использующим ПО собственной разработки.....	12
4.1. Порядок проведения платежей.....	12
4.2. Добавление новых полей в сообщения сервера Киберплат.....	12
4.3. Текущий список полей сообщений сервера.....	13
4.4. Порядок добавления новых кодов ошибок в сообщения сервера Киберплат.....	13

Версии документа

Версия	Дата	Внесённые изменения	Исполнитель
1.0	08.2002г.	Первая версия протокола	Садомсков Д.А.
1.1-1.4.	04.08.2008г.	Расширен список ошибок. Добавлен вариант совершения платежа и проверки номера при неизвестной сумме платежа.	Путилов М.В.
1.5	19.05.2009г.	Добавлено требование обязательного использования HTTPS при обращении к сервисам Киберплат. Внесены незначительные стилевые исправления в текст, не изменяющие технический порядок взаимодействия.	Тимошенко М.В.
1.6	30.09.2009г.	Добавлены опциональные поля в протокол и внесены незначительные исправления в текст, не изменяющие технический порядок взаимодействия.	Бурдынюк А.В.
1.6.1	10.12.2009г.	Расширен список ошибок.	Тимошенко М.В.
1.6.2	28.12.2009г.	Добавлены пояснения и пример запроса с http-заголовком.	Тимошенко М.В.
1.6.3	11.02.2010г.	Уточнены описания полей запросов (AMOUNT и AMOUNT_ALL).	Тимошенко М.В.
1.6.4	27.02.2010г.	Добавлен необязательный параметр ADDINFO на этапе проверки номера и пояснения для результатов проверки номера и проведения платежа.	Тимошенко М.В.
1.6.5	05.05.2010г.	Добавлен раздел рекомендаций для самостоятельной разработки клиентского ПО, расширен список ошибок.	Тимошенко М.В.

1. Общие сведения о протоколе обмена

Программное взаимодействие с платежной системой осуществляется через сеть Internet с использованием TCP/IP в качестве протокола транспортного уровня. Передача данных осуществляется путём обмена сообщениями в рамках протокола прикладного уровня.

Пересылка сообщений производится по протоколу HTTPS, т.е. HTTP ([RFC2068](#), [RFC2616](#)) с обязательным использованием прослойки SSL/TLS. Соединение с сервером устанавливается по порту 443.

Запросы отправляются методом POST. HTTP заголовок “Content-Type” должен содержать “application/x-www-form-urlencoded”. Заголовок “Content-Length” должен содержать верное значение.

Тело запроса передается таким же образом, как и данные, введенные в HTML форму.

В ответе сервера может содержаться заголовок Content-Length. Если клиент использует HTTP версии 1.1 сервер может выдавать ответ используя chunked-кодирование (см. спецификацию HTTP/1.1).

Существует три типа запросов к платежной системе:

- Получение разрешения на платеж (проверка номера телефона/счета на корректность);
- Оплата;
- Проверка состояния платежа.

Для каждого из трех типов запросов используется свой URI (уникальный идентификатор ресурса). Для каждого оператора связи используется своя тройка адресов (в некоторых случаях одна группа адресов используется для нескольких провайдеров).

Если на этапе проверки номера сумма платежа неизвестна – необходимо выполнить 2 проверки номера (1-я на фиктивную сумму, 2-я на реальную). Код сессии платежа должен совпадать с кодом сессии второй проверки номера.

2. Формат сообщений

2.1. Структура сообщения

Сообщения содержат “плоский” текст в ASCII-кодировке, каждый символ кодируется 8-ю битами. Для русских символов используется кодовая страница: Windows-1251. Тело сообщения может состоять из одной или более строк. Переводы строк кодируются парой символов с кодами 0x0D, 0x0A (“\r\n” в Си). Каждая строка имеет следующий вид: “ПАРАМЕТР=ЗНАЧЕНИЕ”. Например, “PHONE=4951234567”.

Перед отправкой запроса. Далее с помощью специализированной библиотеки функций для тела сообщения формируется электронно-цифровая подпись (ЭЦП). Тело сообщения и его ЭЦП оформляются в виде сообщения-контейнера – см. [примеры запросов далее](#).

Тело сообщения с подписью упаковывается специальным образом. Данные операции (подпись и упаковка) выполняются посредством специализированной библиотеки функций. Библиотека доступна для использования разработчиками на языках программирования C/C++, C#, Java, Delphi и распространяется в бинарном виде и в исходных текстах. Поддерживаются следующие операционные системы: GNU Linux 2.4./2.6. (gcc 2.x - 4.x), FreeBSD 4.x/7.x (gcc 2.x - 4.x), Microsoft Windows (msvc). Описание библиотеки включено в состав распространяемого дистрибутива.

Запрос передается в виде значения единственного параметра HTML формы “inputmessage”.

- В поле Content-Type http-заголовка обязательно должно передаваться: application/x-www-form-urlencoded
- В поле User-Agent http-заголовка рекомендуется дополнительно передавать название и версию клиентского ПО, а также коды SD дилера, AP точки и OP оператора. Рекомендованный формат заголовка User-Agent (прямые скобки не передаются):

User-Agent: [название клиентского ПО] ver. [номер версии], SD:
[код дилера], AP: [код точки приёма], OP: [код оператора]

Пример http-запроса с отправкой формы: получение разрешения на платеж

```
POST /cgi-bin/es/es_pay_check.cgi HTTP/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 498

inputmessage=0000037901SM000001180000011800000125%0D%0Aapi17032+
+++++++00017033%0D%0A+++++++0000000%0D%0ABEGIN
%0D%0ASD%3D17031%0D%0AAP%3D17032%0D%0AOP%3D17034%0D%0ASESSION
%3D4b34d1d40000cb80029%0D%0ANUMBER%3D8888888888%0D%0AAMOUNT
%3D11%2E00%0D%0AAMOUNT%5FALL%3D11%0D%0ACOMMENT%3D%0D%0A%0D%0AEND
%0D%0ABEGIN+SIGNATURE%0D
%0AiQBRAWkBAABCiUs00dQBATG5AgDhdZ6RYHykL46QBaAvnHYaY4p0pDjgj04K1
Iyj%0D%0AfSBSvCRpS%2F0EYO9NspuyLeANEQQkkGE
%2F37gUxiPqzAgStXjpsANH%0D%0A%3DvSgb%0D%0AEND+SIGNATURE
```

Ответ возвращается плоским текстом. По своей структуре он строится подобно запросу и подписывается ЭЦП сервера.

Для всех операторов мобильной связи формат запросов идентичен. Для остальных операторов услуг могут добавляться новые поля. Последовательность полей не важна.

Более подробная информация о формате запросов для всех провайдеров услуг, представленных на российской площадке, публикуется на странице:

https://service.cyberplat.ru/cgi-bin/view_stat.utf/help.cgi

2.2. Получение разрешения на платеж (запрос на проверку номера телефона)

Путь:

https://HOST/cgi-bin/KK/KK_pay_check.cgi

здесь:

HOST – адрес сервера;

KK – идентификатор оператора услуг.

Запрос:

SD=XXX – код дилера;

AP=XXX – код точки приема;

OP=XXX – код оператора;

SESSION=XXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии для данной точки приема. Не более 20 символов, X – латинская буква, либо цифра.

Используется для объединения нескольких последовательных запросов различного типа в рамках одной транзакции;

NUMBER=XXXXXXXXXX – номер счёта (или номер телефона) абонента (X – цифра);

ACCOUNT=SSSSSS – номер лицевого счета плательщика (для мобильных операторов пустой);

AMOUNT=XXX.XX – сумма к зачислению (не менее 1 рубля, разделитель – точка);

AMOUNT_ALL=XXX.XX – полная сумма, полученная от плательщика (разделитель – точка);

REQ_TYPE=1 – признак того, что производится проверка номера без дальнейшего проведения платежа. Рекомендуется использовать для фиктивных проверок номера терминалами (когда сумма платежа еще неизвестна)

PAY_TOOL=X – тип оплаты (0 – наличный, 1 – по банковской карте)

TERM_ID=XXXXXX – фактический код точки, отправившей платеж. Используется только агрегаторами для платежей в Билайн.

COMMENT= SSSSSS – комментарий, назначение платежа: только буквы, цифры и пробелы до 64 символов.

Перечисленные выше поля являются обязательными для заполнения, кроме REQ_TYPE, PAY_TOOL, TERM_ID и COMMENT.

Пример запроса с ЭЦП Дилера:

```
0000035401SM000000970000009700000121
api99          00000990
                00000000

BEGIN
SD=199
AP=72
OP=990
SESSION=56567567100010000000
NUMBER=9998887766
ACCOUNT=
AMOUNT=500.00
PAY_TOOL=0
TERM_ID=12345
COMMENT=test

END
BEGIN SIGNATURE
iQBRawkBAAAD3j2r2NwBAeevAf4nvAG4rGAYePHkyVKTt7wffzURhOckd3ctgmG
yQkKWkXh3CLpsbrExsl1VUBl06ih8qHozk2uttXApzHXQXoO
=+pch
END SIGNATURE
```

Ответ:

DATE=DD.MM.YYYY HH:MM:SS – дата и время запроса на оплату;

SESSION= XXXXXXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии, см. выше [описание формата запроса](#);

ERROR=X – код ошибки;

RESULT=X (0 – успех, 1 – ошибка);

OPNAME =SSSSSSSSSSSSSSSSSSSSSS – название Провайдера услуг или домашнего оператора, которому принадлежит телефонный номер (не более 20 символов);

ACCOUNT=XXXXXXXXXX – номер лицевого счета абонента;

ERRMSG=SSSSSSSSSSSSSSSSSSSSSSSS – расшифровка ошибки (пусто в случае удачного платежа);

OPNAME – название домашнего оператора, которому принадлежит телефонный номер (не более 20 символов),

ACCOUNT – номер лицевого счета абонента;

AUTHCODE=XXXXXX – код авторизации у провайдера услуг;

TRANSID=XXXXXX – уникальный идентификатор платежа в КиберПлат;

ADDINFO=XXXXXX – информационное сообщение для плательщика, текст в URL-кодировке. Сообщение, передаваемое в поле ADDINFO, должно выводиться на экран клиентского приложения до отправки запроса на оплату.

Поля OPNAME, ACCOUNT, ERRMSG, AUTHCODE и ADDINFO опциональны и могут не передаваться. Дополнительная информация по использованию опциональных полей в ответах сервера публикуется [на странице онлайн-справки по шлюзам](#).

Если RESULT=0 и ERROR=0 – можно проводить запрос на оплату (шаг 2). При отсутствии ошибки некоторые шлюзы могут оставлять поле ERROR незаполненным. В таких случаях ответ с

RESULT=0 и ERROR= (пустая строка) означает успешный результат проверки номера так же, как и ответ с RESULT=0 и ERROR=0.

Пример ответа сервера с ЭЦП КиберПлат:

```
0000030301SM000000460000004600000121
0J0005          00064182
                00000000

BEGIN
DATE=04.07.2002 12:34:12
SESSION=56567567100010000000
ERROR=0
RESULT=0
OPNAME =ОАО МТС Москва
ACCOUNT=3335566
ERRMSG=

END
BEGIN SIGNATURE
iQBRAwkBAAD6tj1BJ10BAYKxAfsHlQsEFnO2k6ry++W8O8AiJuv4gT+ZVCfZHsKk
c0CbZpP/W3vkljG3xNzMLiqjwbkNuIdwR9Dq7gHmH+ZQMhbT
=LOnP
END SIGNATURE
```

2.3. Запрос на оплату

Путь:

https://URL/cgi-bin/KK/KK_pay.cgi

здесь:

HOST – адрес сервера;

KK – идентификатор оператора услуг.

Выполняется только после получения от сервера разрешения на платеж (т.е. после успешного выполнения шага 1). Если на первом шаге использовался REQ_TYPE=1, то данный запрос вернет ошибку 11.

Разрыв между первым и вторым шагом не может быть больше одного часа, запрос на оплату не может отправляться на следующие сутки.

Запрос:

SD=XXX – код дилера;

AP=XXX – код точки приема;

OP=XXX – код оператора;

SESSION=XXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии,
см. пояснения к параметру SESSION в п. 2.2.;

NUMBER=XXXXXXXXXX – номер телефона плательщика (X – цифра);

ACCOUNT= – номер лицевого счета абонента;

AMOUNT=XXX.XX – сумма к зачислению (не менее 1 рубля, разделитель – точка);

AMOUNT_ALL=XXX.XX – полная сумма, полученная от плательщика (разделитель – точка);

PAY_TOOL=X – тип оплаты (0 – наличный, 1 – по банковской карте);

TERM_ID=XXXXXX – код регистрации, назначенный агрегатором данной точке приема платежей;

RRN=XXXXXX — уникальный идентификатор платежа в системе агента (внешний номер платежа);

DATE=DD.MM.YYYY HH:MM:SS – дата и время платежа в системе агента.

Перечисленные выше поля обязательны для заполнения за исключением PAY_TOOL, TERM_ID, RRN и DATE.

Пример запроса на оплату с ЭЦП Дилера:

```
0000035401SM000000970000009700000121
api99          00000990
                00000000

BEGIN
SD=199
AP=72
OP=990
SESSION=56567567100010000000
NUMBER=9998887766
ACCOUNT=
AMOUNT=500.00
PAY_TOOL=0
TERM_ID=12345

END
BEGIN SIGNATURE
iQBRAwkBAAAD3j2r2NwBAeevAf4nvAG4rGAYAePHkyVKtT7wffzURhOckd3ctgmG
yQkKWkXh3CLpsbrExsllVUBlO6ih8qHozk2uttXApzHXQXoO
=+pch
END SIGNATURE
```

Ответ:

DATE=DD.MM.YYYY HH:MM:SS – дата и время запроса на оплату;

SESSION=XXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии,
см. [пояснения к полю SESSION в запросе разрешения на платёж - п. 2.2.](#)

ERROR=X – код ошибки;

RESULT=X (0 – успех, 1 – ошибка);

TRANSID=XXXXXX – уникальный идентификатор платежа в КиберПлат, см. [пояснение к полю TRANSID в ответе с разрешением на платёж - п. 2.2.](#)

Если RESULT=0 и ERROR=0 – платёж прошёл. При отсутствии ошибок обработки платежа некоторые шлюзы могут оставлять поле ERROR незаполненным. В таких случаях ответ с RESULT=0 и ERROR= (пустая строка) означает успешный платёж так же, как и ответ с RESULT=0 и ERROR=0.

При получении отрицательного ответа на данном этапе рекомендуется проверить состояние платежа.

При возникновении ошибок в программе агента в момент отправки запроса, полном отсутствии ответа от сервера Киберплат, получении ответа, не соответствующего данной спецификации, проверка состояния платежа обязательна.

Пример ответа с ЭЦП сервера КиберПлат:

```

0000030301SM000000460000004600000121
0J0005                00064182
                        00000000

BEGIN
DATE=04.07.2002 12:34:12
SESSION=56567567100010000000
ERROR=0
RESULT=0

END
BEGIN SIGNATURE
iQBRAwkBAAD6tj1BJ10BAYKxAfsHlQsEFnO2k6ry++W8O8AiJuv4gT+ZVCfZHSkK
c0CbZpP/W3vkljG3xNzMLiqjBwkNuIdwR9Dq7gHmH+ZQMhbT
=LOnP
END SIGNATURE

```

2.4. Проверка состояния платежа (запрос статуса)

Путь:

https://URL/cgi-bin/KK/KK_pay_status.cgi

здесь:

HOST – адрес сервера;

KK – идентификатор оператора услуг.

Запрос:

либоSESSION=XXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии,
см. [пояснения к полю SESSION в запросе разрешения на платёж - п. 2.2.](#)**либо**TRANSID=XXXXXX – уникальный идентификатор платежа в КиберПлат, см. [пояснение к полю TRANSID в ответе с разрешением на платёж - п. 2.2.](#)

Пример запроса статуса платежа с ЭЦП Дилера:

```

0000035401SM000000970000009700000121
api99                00000990
                        00000000

BEGIN
SESSION=56567567100010000000

END
BEGIN SIGNATURE
iQBRAwkBAAAD3j2r2NwBAeevAf4nvAG4rGAyAePHkyVKtT7wffzURhOckd3ctgmG
yQkKwKXh3CLpsbrExsllVUBlO6ih8qHozk2uttXApzHXQXoO
=+pch
END SIGNATURE

```

Ответ:

DATE=DD.MM.YYYY HH:MM:SS – дата и время проверки статуса запроса на оплату

SESSION=XXXXXXXXXXXXXXXXXXXX – уникальный идентификатор сессии,
см. [пояснения к полю SESSION в запросе разрешения на платёж - п. 2.2.](#);

ERROR=X – код ошибки;

RESULT=X – состояние платежа;

1 – была только проверка номера (запрос на платёж не поступал);

3 – платёж направлен к оператору связи (обрабатывается);

7 – платеж завершен, можно проверять поле ERROR. Если ERROR=0 – то платеж прошел;

AUTHCODE=XXXXXXXXXXXX – код авторизации провайдера;

TRANSID=XXXXXX – уникальный идентификатор платежа в КиберПлат, см. [пояснение к полю TRANSID в ответе с разрешением на платёж - п. 2.2.](#)

Поля AUTHCODE и TRANSID опциональны и могут не возвращаться либо содержать пустые значения.

В ответах сервера возможны следующие комбинации полей ERROR и RESULT:

- Если RESULT содержит пустую строку, то ERROR содержит код ошибки на запрос получения статуса платежа. Если в поле ERROR сервер возвратил ошибку 11 – это означает, что платеж не зарегистрирован в Киберплат. Необходимо повторить платеж с первого шага с новым номером сессии. Все остальные значения поля ERROR говорят о том, что состояние платежа неизвестно (например, ошибка проверки подписи или сбой БД). Необходимо повторить попытку позднее.
- Если RESULT=1, то платеж в Киберплат не зарегистрирован. Необходимо повторить платеж с первого шага и с новым номером сессии.
- Если $1 < \text{RESULT} < 7$, то платеж находится в состоянии обработки, результат неизвестен. Необходимо повторить попытку позднее.
- Если RESULT = 7, то платеж обработан. В таком случае следует обработать значение поля ERROR. Если ERROR=0, то платеж успешно обработан (проведён). Остальные значения, передаваемые в поле ERROR, представляют собой коды ошибок, возвращаемые в случаях, когда платёж не прошёл.
- Платеж считается успешно завершенным только в том случае, если RESULT = 7 и ERROR=0.

Проверка статуса выполняется до тех пор, пока RESULT не станет равным 7.

Если при проверке статуса установлено, что платеж не прошел, его необходимо повторить с самого первого шага с новым кодом сессии.

Пример ответа на запрос статуса платежа с ЭЦП сервера КиберПлат:

```
0000030301SM000000460000004600000121
0J0005                00064182
                    00000000

BEGIN
DATE=04.07.2002 12:34:12
SESSION=56567567100010000000
ERROR=0
RESULT=7
AUTHCODE=5967596

END
BEGIN SIGNATURE
iQBRawkBAAD6tj1BJwMBAVP6Af0XC+5LDcE/OED4jrKoJjmERFMKOV4NZ1MUkAiM
dKC6eQSXELkVY8fvv1HvTE2G07p4uGoIJ4yLpwYL5tHHPLs7
=mJ0H
END SIGNATURE
```

2.5. Значения поля «ERROR» (коды ошибок сервера Киберплат)

Код результата	Описание
0	Успешное завершение
1	Сессия с таким номером уже существует
2	Неверный код дилера
3	Неверный код точки приема
4	Неверный код оператора
5	Неверный формат кода сессии
6	Неверная ЭЦП
7	Неверный формат или значение суммы вне допустимого диапазона
8	Неверный формат номера телефона
9	Неверный формат номера лицевого счета
10	Неверный формат запроса
11	Сессия с таким номером не существует
12	Запрос сделан с незарегистрированного IP
13	Точка не зарегистрирована у Провайдера услуг
15	Платежи данному оператору не поддерживаются системой
17	Номер телефона не соответствует введенному ранее
18	Сумма платежа не соответствует введенной ранее
19	Номер счета (контракта) не соответствует введенному ранее
20	Платеж находится в состоянии завершения
21	Недостаточно средств для проведения платежа
22	Платеж не принят. Ошибка при переводе средств
23	Неверный номер телефона/счета
223	Контракт клиента не позволяет производить пополнение счета
24	Ошибка связи с сервером провайдера или технологический перерыв в Киберплат
25	Проведение данного типа платежей приостановлено
26	Платежи данного дилера временно заблокированы
27	Операции по счету приостановлены
30	Общая ошибка системы
31	Превышено количество одновременно обрабатываемых запросов (Киберплат)
32	Повторный платеж в течение 60 минут с момента окончания платежа (Киберплат)
33	Превышен максимальный интервал между проверкой номера и платежом (24 часа)
34	Транзакция с таким номером не найдена
35	Ошибка при изменении состояния платежа
36	Неверный статус платежа
37	Попытка обращения в шлюз, отличный от шлюза на предыдущем шаге
38	Неверная дата. Возможно, срок действия платежа истек
39	Неверный счет
40	Карта указанного номинала в системе не зарегистрирована
41	Ошибка сохранения платежа в системе
42	Ошибка при сохранении чека в базе данных

Код результата	Описание
43	Сеанс Вашей работы в системе недействителен (возможно, истек срок действия сессии), попробуйте войти заново
44	Клиент не может работать на этом торговом сервере
45	Отсутствует разрешение на прием платежей в пользу данного провайдера
46	Не удалось завершить ошибочный платеж
47	Сработало временное ограничение прав
48	Ошибка при сохранении данных сессии в базе данных
50	Проведение платежей в системе временно невозможно
51	Не найдены данные в системе
52	Возможно, дилер заблокирован. Текущий статус дилера не позволяет проводить платежи.
53	Возможно, точка приема заблокирована. Текущий статус точки приема не позволяет проводить платежи.
54	Возможно, оператор заблокирован. Текущий статус оператора не позволяет проводить платежи.
55	Тип дилера не позволяет проводить платежи.
56	Ожидалась точка другого типа. Тип точки приема не позволяет проводить платежи
57	Ожидался оператор другого типа. Тип оператора не позволяет проводить платежи.
81	Превышена максимальная сумма платежа.
82	Превышена сумма списаний за день

3. Алгоритм работы программного обеспечения

- Формируется тело запроса [в виде набора параметров и их значений](#);
- С помощью специализированной библиотеки функций для тела запроса формируется ЭЦП, после чего тело запроса и его ЭЦП помещаются в общий контейнер запроса;
- Осуществляется URL-кодирование подписанного запроса: латинские буквы и цифры остаются без изменений, а остальные символы передаются в шестнадцатеричной нотации с префиксом «%», например, «%OD%OA» – перевод строки. Пробелы могут быть заменены символом «+». Внимание! Символы «+» и «=» обязательно должны кодироваться;
- В начале строки подставляется "inputmessage=";
- Сообщение передается на сервер платежной системы в теле HTTP запроса методом POST с указанием заголовков "Content-Length" и "Content-Type" (см. [выше](#));
- В HTTP ответе приходит плоский текст с подписанным ответом платежной системы (без URL-кодирования);
- С помощью специализированной библиотеки функций и открытого ключа сервера КиберПлат проверяется ЭЦП ответного сообщения, пришедшего от сервера;
- В соответствии с настоящим Протоколом интерпретируются значения, переданные в полях ответного сообщения.

4. Рекомендации платёжным агентам, использующим ПО собственной разработки

4.1. Порядок проведения платежей

1. Для осуществления платежа платёжному агенту необходимо последовательно выполнить следующие действия:
 - 1) запрос на возможность проведения платежа (или запрос на проверку номера);
 - 2) запрос на платеж. Отправляется только в после получения положительного ответа на запрос проверки номера;
 - 3) запрос статуса платежа. Выполняется только при отсутствии однозначного ответа на платёжный запрос.
2. Платёжные терминалы и другие устройства, которые проверяют номер Абонента до ввода суммы платежа, должны проводить две проверки номера: первый запрос на условную сумму выполняется до ввода средств Абонентом и второй запрос – на фактическую введенную сумму платежа – после получения средств от Абонента.
 - В первом запросе на проверку номера рекомендуется также указывать дополнительное поле REQ_TYPE со значением 1 (REQ_TYPE=1). После этого второй запрос на проверку номера Абонента и последующий запрос на платёж следует отправлять с тем же номером сессии.
 - Если в первом запросе отсутствует дополнительное поле REQ_TYPE=1 – вторую проверку номера и сам платёж необходимо проводить с новым уникальным номером сессии.
 - Условная сумма, которая указывается при первой проверке номера, обязательно должна быть больше минимальной разрешённой суммы платежа.При несоблюдении перечисленных выше условий платежа по ряду провайдеров проходить не будут.
3. Если программное обеспечение платёжного агента не распознало код ошибки, который вернул сервер Киберплат после запроса на платёж – платёж ни в коем случае нельзя считать успешным. В таких случаях платёжному агенту следует обратиться в службу технической поддержки, запросить разъяснения по полученной ошибке и полный список ошибок сервера CyberPlat®.

4.2. Добавление новых полей в сообщения сервера Киберплат

1. Киберплат оставляет за собой право добавлять новые поля в сообщения сервера.
2. ПО платёжного агента должно без ошибок обрабатывать сообщения сервера с новыми неизвестными полями. Появление новых полей не должно вызывать сбоев в работе агентского ПО или ошибок в интерпретации ответов сервера.
3. Киберплат обязуется информировать платёжных агентов о добавлении новых полей в сообщения сервера и публиковать их описание на web-сайте не менее, чем за 14 дней до ввода в эксплуатацию. В указанный срок платёжный агент должен внести необходимые изменения в собственное агентское ПО и проверить корректность его функционирования на тестовом сервере Киберплат. Информация об изменениях, размещаемая на сервере service.cyberplat.ru, должна регулярно просматриваться платёжным агентом.
4. Если появление новых полей в сообщениях сервера приводит к программным сбоям в клиентском ПО платёжного агента и соответствующая доработка не может быть выполнена до ввода в эксплуатацию обновлений в ПО сервера, ответственное лицо платёжного агента должно обратиться в службу технической поддержки Киберплат. По согласованию с руководством Киберплат сроки ввода в эксплуатацию изменений ПО сервера могут быть перенесены.

4.3. Текущий список полей сообщений сервера

DATE	- дата и время создания сообщения
SESSION	- код сессии платежного агента, получаемый из его запроса
ERROR	- код ошибки (0 – успешный, другие – код соответствующей ошибки)
RESULT	- код результата операции (0 – успешный, 1- ошибочный)
TRANSID	- уникальный код платежа в Киберплат
AUTHCODE	- код авторизации платежа на стороне Поставщика (возвращается частью провайдеров)

Поля, используемые только в платежах МТС

ACCOUNT	- лицевой номер счета абонента МТС (только для МТС-ЕСПП)
ERRMSG	- сообщение об ошибке сервера МТС (только для МТС-ЕСПП)
OPNAME	- код домашнего оператора связи МТС (только для МТС-ЕСПП)

4.4. Порядок добавления новых кодов ошибок в сообщения сервера Киберплат

1. Киберплат оставляет за собой право добавления новых кодов ошибки в сообщениях, возвращаемых платежным сервером.
2. Киберплат обязуется информировать платежных агентов о добавлении новых кодов ошибок и публиковать их описание на web-сайте не менее, чем за 14 дней до ввода их в эксплуатацию. В течение этого срока платежный агент должен внести соответствующие изменения в собственное агентское ПО и проверить корректность его работы с тестовым сервером Киберплат.
3. Киберплат гарантирует, что добавление новых кодов ошибок не изменит логику обработки ответов сервера. Новые коды ошибок назначаются для более детального информирования платежного агента о процессе прохождения платежа на сервере.