

IP2SMS: Описание протокола

IP2SMS: Описание протокола	1
Введение	1
Протокол нижнего уровня	2
Отсылка сообщений	2
Пример отсылки сообщения	5
Получение информации о статусе сообщения	5
Пример запроса статуса	5
Получение суммарной информации о группе сообщений	6
Управление рассылками	6
Автоматическая отправка отчетов о состоянии сообщений	7
Пример реализации на php:	7
Параметры авторизации:	8

Введение

Данный документ является описанием протокола взаимодействия программного обеспечения компаний-партнеров с IP2SMS-платформой в рамках проекта IP2SMS. Протокол обеспечивает программному обеспечению партнеров следующие возможности:

- отправку коротких сообщений (SMS) мобильным абонентам
- получение информации о статусе отосланных сообщений

В качестве тела отсылаемых SMS могут использоваться как текст, так и двоичные данные в разных форматах (например, EMS, Nokia CLI и т.п.)

Существенными целями при разработке данного протокола являлись:

- гибкость и расширяемость
- простота реализации
- отсутствие необходимости в сложном программном обеспечении со стороны партнеров

Исходя из указанных целей, в качестве основы для протокола был выбран обмен XML-сообщениями через протокол HTTP. Использование языка XML обеспечивает необходимую гибкость и расширяемость. Наличие большого количества готовых средств генерации и обработки XML, а также простая структура протокола делают реализацию клиентской части не очень сложной.

Протокол опирается на стандартную модель "клиент-сервер", причем в качестве клиента выступает программное обеспечение партнера, а в качестве сервера - IP2SMS-платформа. В последующих разделах мы будем для краткости использовать термины "клиент" и "сервер" именно в этом смысле.

В случае отсылки сообщений, клиент посылает серверу запрос, содержащий как собственно сообщение, так и сопутствующую информацию. В ответ, сервер инициирует отсылку сообщения и возвращает статус завершения операции. Также сервер возвращает уникальный идентификатор сообщения для последующих ссылок со стороны клиента. В случае выяснения статуса отосланного сообщения, клиент посылает серверу запрос, содержащий полученный ранее уникальный идентификатор сообщения. В ответ, сервер возвращает клиенту текущий статус сообщения.

Следующий раздел данного документа содержит описание взаимодействия клиента и сервера на нижнем уровне (уровне HTTP). Последующие два раздела подробно описывают два основных типа запросов.

Протокол нижнего уровня

Как уже говорилось ранее, в качестве транспортного уровня для описываемого протокола был выбран протокол HTTP. Соответственно, на нижнем уровне все запросы клиента к серверу выглядят как обычные HTTP запросы, использующие метод POST, в которых тело запроса является XML-текстом. В результате выполнения запроса, IP2SMS-платформа возвращает XML-текст, содержащий результат. Тип возвращаемого результата (Content-Type) всегда `text/xml`.

Отсылка сообщений

Для отсылки сообщения, клиент передает серверу XML-запрос следующего вида:

Отсылка одиночного сообщения:

```
<message>
  <service id="single" start="время начала" validity="период актуальности"
source="исходящий номер"/>
  <to>номер получателя</to>
  <body content-type="тип содержимого"
encoding="кодировка"
тело
</body>
</message>
```

ВНИМАНИЕ: запрещается использовать режим отправки одиночных сообщений «single» для отправки массовых рассылок — для этого необходимо использовать режимы «bulk» или «individual».

Отсылка одного текста на несколько номеров:

```
<message>
  <service id="bulk" start="время начала" validity="период актуальности"
rate="скорость отправки" desc="название рассылки" source="исходящий номер"
uniq_key="уникальный ключ рассылки"/>
  <to>number1</to>
  <to>number2</to>
  ...
  <to>numberN</to>
  <body content-type="тип содержимого" encoding="кодировка">тело</body>
</message>
```

ВНИМАНИЕ: запрещается использовать режим отправки массовых рассылок «bulk» и «individual» для создания рассылок состоящих из одного сообщения – для этого необходимо использовать режим «single».

Отсылка индивидуального текста на несколько номеров:

```
<message>
  <service id="individual" start="время начала" validity="период
актуальности" rate="скорость отправки" desc="название рассылки"
source="исходящий номер" uniq_key="уникальный ключ рассылки"/>
  <to>number1</to>
  <body content-type="тип содержимого1" encoding="кодировка1">тело1</body>
  <to>number2</to>
  <body content-type="тип содержимого2" encoding="кодировка2">тело2</body>
  ...
  <to>numberN</to>
  <body content-type="тип содержимогоN" encoding="кодировкаN">телоN</body>
</message>
```

Здесь:

- **время начала** – время начала рассылки в стандартном формате. Может задаваться в абсолютном виде («Mon, 02 Apr 2007 11:58:24 +0300») или относительно текущего времени («+2 hour», «+1 day», «+3 hour 20 min»). Если параметр пропущен - рассылка начинается сразу после получения запроса.
- **период актуальности** – время актуальности доставки сообщения в стандартном формате. Может задаваться в абсолютном виде («Mon, 02 Apr 2007 11:58:24 +0300») или относительно времени отправки сообщения («+2 hour», «+1 day», «+3 hour 20 min»). Если параметр пропущен, значение по умолчанию равно 2 часам от момента отправки сообщения абоненту.
- **скорость отправки** – ограничение скорости отправки сообщений абонентам. Задается в количестве сообщений, отправляемых в минуту. Может принимать значения от 1 до 100. Если параметр пропущен – рассылка выполняется с максимально возможной скоростью.
- **название рассылки** – текстовое имя для рассылки (необязательный параметр). Используется в web-интерфейсе.
- **исходящий номер** – исходящий номер, с которого производится рассылка.
- **уникальный ключ рассылки** – случайное число от 0 до 2^{31} , не изменяющееся при нескольких попытках отправки одной рассылки. Служит для предотвращения добавления одной рассылки более одного раза при случайных сбоях связи;
- **номер получателя** - номер получателя – Мобильного абонента. Номер должен задаваться в полном международном формате, состоящем из знака + и двенадцати цифр. Например: +380671234567
- **тип содержимого** - строка, задающая тип содержимого отсылаемого сообщения. Способ задания строки совпадает с определенным стандартом MIME. В данный момент могут использоваться следующие типы содержимого:
 - **text/plain** - текстовое сообщение. При использовании этого типа, атрибут **encoding** игнорируется. Подразумевается, что тело сообщения содержит исключительно текст в кодировке UTF-8.
- **кодировка** - задает кодировку тела сообщения, если тело сообщения двоичное. Для текстовых сообщений (*тип сообщения* - text/plain) данный параметр можно опустить, либо установить равным plain. Допустимые значения *кодировки* - следующие:

- plain - кодирование отсутствует. Данный тип допустим только для текстовых сообщений в UTF-8
- base64 - тело сообщения закодировано как BASE64 (см. стандарт MIME)

В результате обработки запроса, сервер возвращает клиенту результат следующего вида:

В случае service id="single":

```
<status id="ид сообщения" date="дата и время">
  <state
    error="сообщение об ошибке">
      статус
    </state>
</status>
```

В случае service id="bulk" или "individual":

```
<status groupid='ид группы сообщений' date='дата и время'>
  <id>ид сообщения1</id><state error="сообщение об ошибке">статус1</state>
  <id>ид сообщения2</id><state error="сообщение об ошибке">статус2</state>
  ...
  <id>ид сообщенияN</id><state error="сообщение об ошибке">статусN</state>
</status>
```

Здесь:

- **ид группы сообщений** - уникальный идентификатор группы сообщений. Может использоваться клиентом для последующих запросов суммарной статистики для данной группы сообщений.
- **ид сообщения** - уникальный идентификатор сообщения. Может использоваться клиентом для последующих запросов статуса сообщения. Данный атрибут может отсутствовать, если сообщение не может быть доставлено (ошибка XML, ошибка формата, неверный IP-адрес, несуществующий идентификатор услуги и т.п.)
- **дата и время** - дата и время, для которых указывается статус. Следует особо отметить, что дата и время, установленные на IP2SMS-платформе могут отличаться от установленных на компьютере клиента
- **сообщение об ошибке** - сообщение об ошибке, если таковая произошла. Сообщения представляют собой текстовые строки и предназначены для анализа человеком, а не программой. В случае отсутствия несомненной ошибки данный атрибут опускается
- **статус** - собственно статус сообщения. Перечень возможных статусов может расширяться. На данный момент это:
 - Accepted - сообщение принято IP2SMS-платформой, но попытка доставки еще не предпринималась
 - Enroute - предпринимаются попытки доставить сообщение, однако, оно еще не доставлено
 - Delivered - сообщение доставлено получателю
 - Expired - исчерпан лимит времени на попытки доставить сообщение; последующие попытки предприниматься не будут
 - Deleted - сообщение принудительно удалено из системы администратором
 - Undeliverable - сообщение по тем или иным причинам не может быть доставлено получателю (например, попытка доставить на несуществующий телефонный номер)
 - Rejected - сообщение отвергнуто из-за ошибок в нем (нарушение формата, попытка отправить сообщение пределы украинских операторов и т.п.)

- o Unknown - состояние сообщения неизвестно.

Если статус отличен от Accepted, Enroute и Delivered, атрибут error тэга state может содержать уточняющую информацию об ошибке

Пример отсылки сообщения

Ниже приведены примеры запроса на отправку сообщения и ответа IP2SMS-платформы.
Запрос:

```
<message>
<service id="single"/>
<to>+380671234567</to>
<body content-type="text/plain">
This is a sample message
</body>
</message>
```

Ответ:

```
<status id="3806712345671174984921384" date="Wed, 28 Mar 2007 12:35:00 +0300">
<state>Accepted</state>
</status>
```

Получение информации о статусе сообщения

Для получения информации о статусе сообщения, клиент должен послать серверу запрос следующего вида:

```
<request id="ид сообщения">status</request>
```

где *ид сообщения* - полученный в результате отправки идентификатор сообщения (см. выше)

В ответ сервер посылает клиенту информацию о статусе сообщения в описанном выше виде. Единственным отличием является возможность появления *статуса* not found означающего, что сообщение с указанным идентификатором не найдено.

Пример запроса статуса

Ниже приведен пример запроса статуса сообщения и ответа IP2SMS-платформы.

Запрос:

```
<request id="3806712345671174984921384">status</request>
```

Ответ:

```
<status id="3806712345671174984921384" date="Wed, 28 Mar 2007 12:35:00 +0300">
<state>Delivered</state>
</status>
```

Получение суммарной информации о группе сообщений

Для получения суммарной информации о группе сообщений, клиент должен послать серверу запрос следующего вида:

```
<request groupid="ид группы сообщений">status</request>
```

где *ид группы сообщений* - полученный в результате отправки идентификатор группы сообщений (см. выше).

В результате сервер дает ответ в следующей форме:

```
<status groupid="ид группы сообщений" date="дата" desc="имя рассылки"
state="состояние рассылки" reports="состояние приема отчетов">
<total>количество</total>
<queued>количество</queued>
<accepted>количество</accepted>
<enroute>количество</enroute>
<delivered>количество</delivered>
<expired>количество</expired>
<undeliverable>количество</undeliverable>
<unknown>количество</unknown>
</status>
```

Некоторые тэги могут отсутствовать, если соответствующее количество равно нулю.

Поле «состояние рассылки» содержит информацию о текущем состоянии рассылки и может принимать следующие значения:

- waiting – рассылка еще не началась;
- sending – рассылка в процессе;
- sent – рассылка завершена;
- paused – рассылка приостановлена;
- canceled – рассылка частично отменена.

Поле «состояние приема отчетов» содержит информацию о текущем состоянии приема отчетов о доставке сообщений от операторов. Оно появляется только когда опрака сообщений завершена (state="sent") и может принимать следующие значения:

- waiting – период актуальности сообщений еще не прошел, система ожидает отчеты о доставке сообщений;
- completed – прием отчетов о доставке завершен.

Управление рассылками

Рассылка, находящаяся в системе может быть приостановлена, продолжена или отменена пользователем. Для управления рассылкой пользователь должен подать команду следующего вида:

```
<request groupid="ид группы сообщений">команда</request>
```

В ответ система выдаст отчет:

```
<status groupid="ид группы сообщений" date="дата" state="состояние"> </status>
```

если операция прошла успешно либо:

```
<status groupid="ид группы сообщений" date="дата" error="описание  
ошибки"></status>
```

в случае, если команду выполнить не удалось.

команда – может принимать значения pause, resume или cancel;

состояние может быть paused, resumed, canceled или deleted.

При команде cancel система анализирует состояние рассылки и, если рассылка еще не началась – она удаляется из системы полностью (deleted), в противном случае удаляется часть рассылки, которая еще не была отправлена абонентам (canceled). Команда cancel может подаваться только для рассылок, которые находятся в состоянии waiting или paused.

Автоматическая отправка отчетов о состоянии сообщений

Для работы в этом режиме клиент предоставляет url собственной системы, на который будет отправляться отчет при установлении окончательного состояния отправленного сообщения. Вид отчета:

```
<status id="3806712345671174984921384" date="Wed, 28 Mar 2007 12:35:00 +0300">  
<state error="текст ошибки">состояние</state>  
</status>
```

Где "состояние" может принимать значения "undeliverable", "expired", "unknown" "delivered". Поле "текст ошибки" может содержать уточняющую информацию в случае установки состояния "undeliverable".

Система клиента должна дать подтверждение в следующей форме:

```
<status>accepted</status>
```

Пример реализации на php:

```
<?php  
$from="TEST_NUMBER";  
$to="380987654321";  
$start="14:00";  
$text="Тестовое сообщение";  
  
$xml="<message><service id='single' validity='+2 hour' source='$from'  
start='$start'/><to>$to</to><body content-type='plain/text'  
encoding='plain'>$text</body></message>";  
  
$answ=post_request($xml, 'http://bulk.startmobile.com.ua/clients.php', 'super-  
login', 'mega-password');  
  
function post_request($data, $url, $login, $pwd)  
{  
    $credent = sprintf('Authorization: Basic  
%s',base64_encode($login.":".$pwd) );  
    $params=array( 'http'=>array( 'method'=>'POST', 'content'=>$data,  
'header'=>$credent) );  
    $ctx = stream_context_create($params);  
    $fp=@fopen($url, 'rb', FALSE, $ctx);  
    if ($fp)
```

```
{  
    $response = @stream_get_contents($fp);  
}else  
{  
    $response = false;  
}  
  
return $response;  
}  
?>
```

Параметры авторизации:

Логин и пароль аналогичны, что и для доступа в web интерфейс системы.