



## **NetSDS SMPP Server 1.2**

**Техническое задание**

[www.netstyle.com.ua](http://www.netstyle.com.ua)

+380 44 594-9641

+380 44 221-1286

## Содержание

Общее описание.....	3
Функциональная спецификация.....	3
Обработка событий протокола SMPP.....	4
Поддерживаемые команды SMPP 3.4.....	4
Пример SMPP-сессии.....	5
Заголовок PDU SMPP.....	5
Установка соединения — bind_transceiver.....	6
Подтверждение установки соединения — bind_transceiver_resp.....	6
Запрос завершения соединения — unbind.....	6
Подтверждение завершения соединения — unbind_resp.....	6
Отправка МТ сообщения — submit_sm.....	6
Подтверждение отправки МТ сообщения — submit_sm_resp.....	7
Доставка МО сообщения или DLR — deliver_sm.....	7
Подтверждение доставки МО сообщения — deliver_sm_resp.....	8
Проверка соединения — enquire_link.....	8
Подтверждение проверки соединения — enquire_link_resp.....	8
Ответ на некорректную команду SMPP — generic_nack.....	8
Статусы выполнения команд SMPP.....	8
Уведомления о доставке.....	8
Аутентификация клиентов SMPP.....	9
Обработка очередей сообщений.....	10
Внутренний формат сообщений.....	10
Параметры сериализованных сообщений JSON.....	10
Администрирование NetSDS SMPP Server.....	11
Запуск.....	11
Остановка.....	11
Изменение конфигурации.....	11
Мониторинг.....	11
Требования к системному ПО.....	11

## Общее описание

Платформа **NetSDS SMPP Server** поддерживает подключение клиентского программного обеспечения по протоколу SMPP (Short Message Peer-to-Peer) версии 3.4 в режиме трансивера. В частности, описываемый SMPP интерфейс позволяет устанавливать соединение, получать и отправлять SMS, получать уведомления о доставке сообщений, удалять сообщения из очереди, проверять работоспособность соединения.

В документе используются следующие соглашения о наименованиях:

- **ESME** — клиентское приложение
- **CPA** — клиентский интерфейс платформы NetSDS
- **DLR** — уведомление о доставке
- **SM** — короткое сообщение (Short Message)
- **MO** — событие, инициируемое абонентским терминалом (Mobile Originated)
- **MT** — событие, инициируемое сервером (Mobile Terminating)

**Детальную информацию про упоминаемые команды протокола и их параметры можно получить в официальной спецификации SMPP v 3.4.**

## Функциональная спецификация

Функциональность	Статус	Комментарии
<b>Поддержка функций протокола SMPP 3.4</b>		
Поддержка протокола SMPP	✓	Версия 3.4
Режим transceiver	✓	
Режим transmitter/receiver	✗	
Поддержка передачи MT сообщений	✓	
Поддержка передачи MO сообщений	✓	
Поддержка передачи уведомлений о доставке (DLR)	✓	
Поддержка мониторинга канала (enquire_link)	✓	
<b>Аутентификация</b>		
SQL СУБД (MySQL, PostgreSQL, SQLite, Oracle)	✓	При наличии драйвера для Perl DBI
LDAP	✗	В следующих версиях
RADIUS	✗	В следующих версиях
<b>Поддержка форматов передаваемых сообщений</b>		
Текстовые сообщения в кодировке GSM 03.38	✓	
Текстовые сообщения в кодировке UCS2	✓	
Бинарные 8-битные сообщения	✓	
Длинные «склеенные» сообщения	✓	С использованием UDH,

		согласно GSM 03.40
Поддержка передачи UDH	✓	
Поддержка установки message class	✓	
Передача MWI (mail waiting indication)	✗	
<b>Статистика и интеграция с биллинговыми системами</b>		
Поддержка уведомлений о доставке SMS (DLR)	✓	
Запись CDR для обработки биллинговыми системами	✓	Запись в файл
Передача Service Code для обработки SMSC	✗	

## Обработка событий протокола SMPP

В этом разделе описывается подмножество команд протокола SMPP, поддерживаемых платформой **NetSDS SMPP Server**. Использование этих команд позволяет клиентскому ПО принимать и отправлять сообщения SMS, а также принимать уведомления о статусе доставки SMS.

### Поддерживаемые команды SMPP 3.4

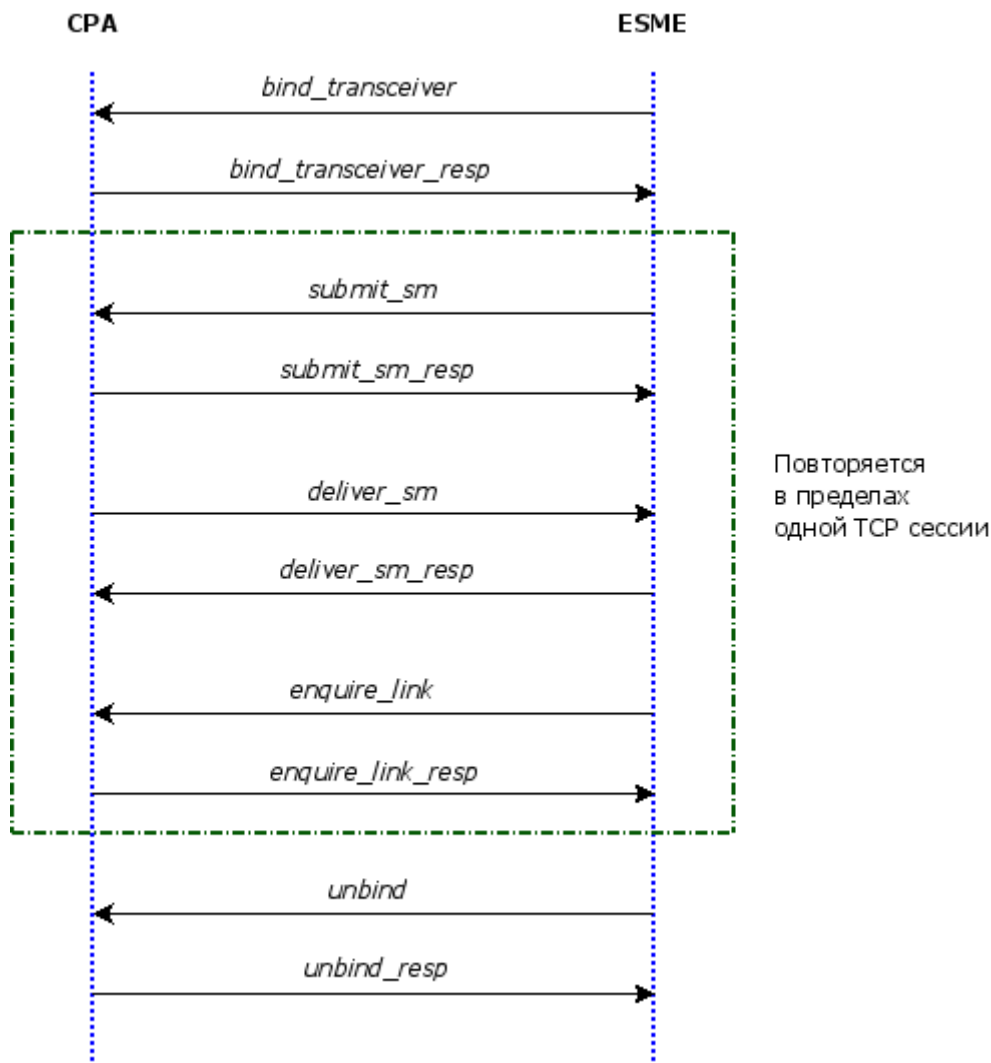
В этом разделе спецификации перечислен список команд протокола SMPP 3.4, поддерживаемых платформой **NetSDS SMPP Server**.

Тип SMPP PDU	PDU command_id	Запрос от ESME	Запрос от CPA
bind_transceiver	0x00000009	Да	Нет
bind_transceiver_resp	0x80000009	Нет	Да
unbind	0x00000006	Да	Да
unbind_resp	0x80000006	Да	Да
submit_sm	0x00000004	Да	Нет
submit_sm_resp	0x80000004	Нет	Да
deliver_sm	0x00000005	Нет	Да
deliver_sm_resp	0x80000005	Да	Нет
enquire_link	0x00000015	Да	Да
enquire_link_resp	0x80000015	Да	Да
generic_nack	0x80000000	Да	Да

Данный список включает все необходимые команды протокола SMPP для обмена сообщениями между ESME и CPA API, но может быть расширен в случае необходимости реализации специфичных для конкретной инсталляции функций (например, для поддержки USSD или Subscriber Registration Alarm).

## Пример SMPP-сессии

В этом разделе приводится пример типичной SMPP-сессии между клиентским программным обеспечением (ESME) и **NetSDS SMPP Server**.



После установки TCP-соединения ESME отправляет PDU с командой ***bind\_transceiver***, в ответ на которую CPA API отправляет PDU с командой ***bind\_transceiver\_resp***, после чего сессию SMPP можно считать установленной.

В процессе взаимодействия по протоколу SMPP ESME может отправлять команды ***submit\_sm***, ***enquire\_link***, в ответ на которые CPA API обязан возвращать PDU с командами ***submit\_sm\_resp***, и ***enquire\_link\_resp***, соответственно.

Для получения сообщений (MO SM) и уведомлений о доставке ранее отправленных SMS, ESME должен принимать от CPA API PDU с командой ***deliver\_sm*** и корректно в ответ отправлять PDU с командой ***deliver\_sm\_resp***.

Для завершения сессии сторона, инициирующая завершение, должна отправить PDU с командой *unbind*, после чего дождаться PDU с ***unbind\_resp*** и разорвать TCP-сессию.

## Заголовок PDU SMPP

Любая команда SMPP содержит заголовок со следующими полями:

Имя параметра	Описание
---------------	----------

<b>command_length</b>	Размер всего PDU в байтах
<b>command_id</b>	Идентификатор команды SMPP
<b>command_status</b>	Статус выполнения команды в ответных PDU
<b>sequence_number</b>	Идентификатор последовательности в рамках SMPP сессии для вычисления взаимосоответствия PDU.

### Установка соединения — **bind\_transceiver**

Эта команда отправляется ESME сразу же после инициации TCP сессии для инициализации соединения по протоколу SMPP. В **NetSDS SMPP Server** все параметры PDU кроме **system\_id** и **password** являются необязательными и в большинстве случаев их содержимое игнорируется.

Имя параметра	Описание
<b>system_id</b>	Имя системы ESME
<b>password</b>	Пароль доступа
<b>interface_version</b>	Версия интерфейса (поддерживается 0x34)
<b>addr_ton</b>	Тип номера ESME (см. п. 5.2.5 спецификации SMPP)
<b>addr_npi</b>	План нумерации ESME (должен быть 00000001)
<b>address_range</b>	Диапазон номеров, обслуживаемых ESME (игнорируется)

### Подтверждение установки соединения — **bind\_transceiver\_resp**

Эта команда отправляется в ответ на **bind\_transceiver** и служит для подтверждения успешного соединения либо сообщения ошибки соединения. В случае успешного соединения также передается имя платформы.

Имя параметра	Описание
<b>system_id</b>	Имя системы CPA (должно быть «NETSDS»)

### Запрос завершения соединения — **unbind**

Команда **unbind** отправляется стороной, иницирующей разрыв SMPP сессии. Также допускается закрытие сессии просто по факту разрыва TCP соединения, однако такое поведение является нежелательным, нарушая стандартный процесс взаимодействия в рамках SMPP 3.4.

Команда содержит только заголовок SMPP PDU.

### Подтверждение завершения соединения — **unbind\_resp**

Команда содержит только заголовок SMPP PDU.

### Отправка MT сообщения — **submit\_sm**

Имя параметра	Описание
<b>service_type</b>	Тип сервиса (используется для тарификации)
<b>source_addr_ton</b>	Тип номера ESME (см. п. 5.2.5 спецификации SMPP)
<b>source_addr_npi</b>	План нумерации ESME (должен быть 00000001)
<b>source_addr</b>	Сервисный адрес (короткий номер, А-номер)
<b>dest_addr_ton</b>	Тип номера абонента (обычно, 00000001)

<b>dest_addr_npi</b>	План нумерации абонента (должен быть 00000001)
<b>destination_addr</b>	Адрес абонента (MSISDN)
<b>esm_class</b>	См. п. 5.2.12 спецификации SMPP
<b>protocol_id</b>	См. п. 5.2.12 спецификации SMPP (обычно 0x00000000)
<b>priority_flag</b>	От 0 (низший) до 3 (высший) приоритет
<b>schedule_delivery_time</b>	Время доставки
<b>validity_period</b>	Время жизни сообщения
<b>registered_delivery</b>	См. п 5.2.17 спецификации SMPP (обычно 0x00000001)
<b>replace_if_present_flag</b>	Не поддерживается
<b>data_coding</b>	См. п 5.2.19 спецификации SMPP
<b>sm_default_msg_id</b>	Не поддерживается
<b>sm_length</b>	Длина сообщения
<b>short_message</b>	Содержимое сообщения (User Data)

### Подтверждение отправки MT сообщения — `submit_sm_resp`

В ответ на `submit_sm` CPA API передает PDU с командой `submit_sm_resp`, содержащей идентификатор сообщения, назначенный платформой **NetSDS SMPP Server**.

### Доставка MO сообщения или DLR — `deliver_sm`

Данная команда служит для передачи от CPA API к ESME сообщений.

Имя параметра	Описание
<b>service_type</b>	Тип сервиса (используется для тарификации)
<b>source_addr_ton</b>	Тип номера абонента (обычно, 00000001)
<b>source_addr_npi</b>	План нумерации абонента (должен быть 00000001)
<b>source_addr</b>	Адрес абонента (MSISDN)
<b>dest_addr_ton</b>	Тип номера ESME (см. п. 5.2.5 спецификации SMPP)
<b>dest_addr_npi</b>	План нумерации ESME (должен быть 00000001)
<b>destination_addr</b>	Сервисный адрес (короткий номер)
<b>esm_class</b>	См. п. 5.2.12 спецификации SMPP
<b>protocol_id</b>	См. п. 5.2.12 спецификации SMPP (обычно 0x00000000)
<b>priority_flag</b>	От 0 (низший) до 3 (высший) приоритет
<b>schedule_delivery_time</b>	Время доставки
<b>validity_period</b>	Время жизни сообщения
<b>registered_delivery</b>	См. п 5.2.17 спецификации SMPP (обычно 0x00000001)
<b>replace_if_present_flag</b>	Не поддерживается
<b>data_coding</b>	См. п 5.2.19 спецификации SMPP
<b>sm_default_msg_id</b>	Не поддерживается
<b>sm_length</b>	Длина сообщения
<b>short_message</b>	Содержимое сообщения (User Data)

### Подтверждение доставки МО сообщения — deliver\_sm\_resp

Данная команда является ответом, подтверждающим успешный прием МО SM либо DLR на стороне ESME.

Имя параметра	Описание
message_id	Выставляется в NULL

### Проверка соединения — enquire\_link

Команда содержит только заголовок SMPP PDU.

### Подтверждение проверки соединения — enquire\_link\_resp

Команда содержит только заголовок SMPP PDU.

### Ответ на некорректную команду SMPP — generic\_nack

Данная команда отправляется в ответ на PDU с некорректной длиной либо неизвестным платформе идентификатором команды.

Команда содержит только заголовок SMPP PDU.

### Статусы выполнения команд SMPP

Числовые значения статусов указаны в п. 5.1.3 спецификации SMPP 3.4.

Код статуса	Описание
ESME_ROK	Успешное выполнение
ESME_RINVCMDID	Некорректный идентификатор команды
ESME_RALYBND	ESME уже подключен
ESME_RSYSERR	Ошибка платформы
ESME_RINVSRCADR	Некорректный исходящий адрес
ESME_RINVDSTADR	Некорректный входящий адрес
ESME_RINVMGID	Некорректный идентификатор сообщения
ESME_RINVPASWD	Некорректный пароль
ESME_RINVSERTYP	Некорректный тип сервиса

### Уведомления о доставке

Для уведомлений о статусе доставки используется следующий формат:

**id:ID sub:NN dlvrdr:NN submit date:DATE done date:DATE stat:STATE**

#### Значимые поля:

- **id** — идентификатор сообщения (UUID)
- **done date** — дата и время доставки (YYYYMMDDhhmm)
- **state** — статус доставки (DELIVRD, EXPIRED, DELETED, UNDELIV, ACCEPTED, UNKNOWN, REJECTED)

## Аутентификация клиентов SMPP

Для аутентификации клиентов **NetSDS SMPP Server** использует подключение к любой реляционной СУБД, поддерживающей язык запросов SQL. Шаблон SQL запроса задается в файле конфигурации, в секции **auth** и включает два параметра, в которые помещаются имя пользователя (**system-id**) и пароль (**password**).

Для успешной аутентификации требуется, чтобы при вызове SQL запроса с указанными параметрами СУБД вернула одну или более строк.

Пример:

### SQL:

```
-- Creating SQL table for SMPP users
CREATE TABLE users (
    id integer,
    login varchar(16),
    passwd varchar(16),
    active integer
)
```

### Конфигурация:

```
# SQL authentication backend
<auth>
    dsn = db:Pg:dbname=authdb
    db-user = smpp
    db-password = secret
    query = select user from users where login=? and passwd=? and active=1
</auth>
```

Таким образом, **NetSDS SMPP Server** позволяет производить аутентификацию клиентов, используя фактически любую СУБД с языком запросов SQL, для которой существует реализация драйвера DBI.

## Обработка очередей сообщений

Для взаимодействия с приложениями бизнес-логики **NetSDS SMPP Server** использует подсистему управления очередями **MemcacheQ**<sup>1</sup>, работающую по протоколу **memcache**<sup>2</sup>. Данный протокол базируется на TCP, но также может применяться взаимодействие через протокол передачи датаграмм UDP или unix socket.

1. Все сообщения, которые успешно приняты платформой от ESME, преобразуются в стандартное внутреннее представление и помещаются в очередь **q\_mt**
2. Сообщения для передачи к ESME выбираются из очереди **q\_mo\_SYSID**, где SYSID соответствует system-id, выданному клиенту при регистрации.

Таким образом, для всех MT сообщений используется одна очередь, из которой эти сообщения могут быть получены приложением, отвечающим за их дальнейшую обработку.

**Все MO SM и DLR помещаются в отдельные очереди для каждого ESME, что позволяет снизить нагрузку на платформу за счет обработки только событий для активных клиентов.**

## Внутренний формат сообщений

В качестве внутреннего формата для обмена сообщениями применяется JSON, сконвертированный в строку Base64 для передачи по протоколу Memcache. Каждое сообщение представляется структурой (ассоциативным массивом) приблизительно следующего вида:

```
{
  "type": "mo",
  "from": "380672206770"
  "to": "12345",
  "text": "message text here",
  "coding": "gsm0338"
}
```

При этом, параметр **type** задает тип сообщения и может принимать следующие значения:

- **mo** — MO SM (сообщение от абонента к сервису)
- **mt** — MT SM (сообщение от сервера к абоненту)
- **dlr** — DLR (уведомление о доставке)

В зависимости от типа сообщения, содержимое сериализованного массива может включать различные наборы параметров.

### Параметры сериализованных сообщений JSON

Данный раздел описывает поддерживаемые параметры JSON-представления сообщений.

Поле	Содержимое	Формат
<b>client</b>	Идентификатор клиента (SMPP system-id)	Строка ASCII
<b>id</b>	Идентификатор сообщения	UUID
<b>created</b>	Дата создания сообщения	ГГГГ-ММ-ДД ЧЧ:ММ:СС

1 <http://memcachedb.org/memcacheq/> - официальный сайт MemcacheQ

2 <http://code.sixapart.com/svn/memcached/trunk/server/doc/protocol.txt> — описание протокола memcache

<b>from</b>	Адрес источника сообщения	MSISDN или Alphanumeric
<b>to</b>	Адрес получателя сообщения	MSISDN или Alphanumeric
<b>udh</b>	User Data Headers (заголовок SM)	Шестнадцатиричная строка
<b>ud</b>	User Data (содержимое SM)	Шестнадцатиричная строка
<b>coding</b>	Кодировка сообщения	Строка с одним из вариантов: <b>gsm0338, 8bit, ucs2</b>
<b>mclass</b>	Класс сообщения, если есть	Число от 0 до 3. См. спецификацию GSM 03.38
<b>priority</b>	Приоритет доставки для передачи SMSC	Число от 0 до 3.
<b>validity</b>	Время жизни сообщения, если требуется явное ограничение. В UTC.	ГГГГ-ММ-ДД ЧЧ:ММ:СС
<b>deferred</b>	Время отложенной доставки сообщения, если требуется. В UTC.	ГГГГ-ММ-ДД ЧЧ:ММ:СС
<b>text</b>	Текстовое представление содержимого SMS, если допустимо.	Строка в кодировке UTF-8
<b>dlr_state</b>	Статус доставки сообщения	Число согласно спецификации SMPP 3.4 п 5.2.28
<b>dlr</b>	Текстовое представление DLR	Строка ASCII
<b>network_error</b>	Код ошибки GSM-сети	Число
<b>register_delivery</b>	Требуется ли уведомление о доставке	0, если не требуется 1, если требуется

## Требования к производительности

1. На незагруженной системе с CPU Intel Xeon 3300, 1GB RAM, SATA II HDD **NetSDS SMPP Server** должен обрабатывать не менее 1000 сообщений в секунду.
2. **NetSDS SMPP Server** на вышеописанной конфигурации должен корректно обрабатывать не менее 100 одновременных клиентских подключений по протоколу SMPP 3.4.
3. Система должна корректно работать при количестве необработанных сообщений в очереди до 5,000,000.

## Требования к системному ПО

- Операционная система: Linux, FreeBSD
- Интерпретатор Perl: 5.8.X
- Сервер управления очередями MemcacheQ
- BerkeleyDB 4.7
- Возможность инсталлировать пакеты из CPAN

Рекомендуется использование системы мониторинга приложений Monit для обеспечения бесперебойной работы всех приложений и отслеживания критичной нагрузки на ресурсы системы.