

SMPP API Guidelines

Contents

1. SMPP API Account Usage Instructions
 - 1.1. Preliminaries
 - 1.2. Binding
 - 1.3. Message submission
 - 1.4. Receiving incoming messages
 - 1.5. Unbinding
2. SMPP API Implementation Guidelines
 - 2.1. Receiver Binding
 - 2.2. SMSC/ESME Disconnection
 - 2.3. Re-Connect Behaviour of ESME Applications
 - 2.3.1. Definitions

1. SMPP API Account Usage Instructions

1.1. Preliminaries

Applications (or ESMEs in SMPP terminology) connecting to [LINK Mobility UK's](#) messaging infrastructure to use our SMPP API must support at least SMPP v3.3. If your firewall policy prevents access from your network then your firewall should be configured to allow access to the hosts and ports listed in the SMPP API configuration document provided by LINK Mobility UK.

1.2. Binding

SMPP receiver and transmitter binds or a transceiver bind must be made to LINK Mobility UK in order for your application to communicate using LINK Mobility UK's messaging infrastructure when using the SMPP API. At all times an SMPP receiver bind must be maintained if you wish to receive delivery receipts as and when they are generated. Transmitter binds can remain permanently connected or may be bound and unbound as required.

Multiple binds for receiver and transmitter may be made from the same or different instances of your application on one or more hosts.

No more than five (5) bindings for the transmitter or receiver are permitted without prior agreement.

When establishing a receiver, transmitter or transceiver bind using the SMPP API your application **MUST** use the SMPP configuration parameter values supplied by LINK Mobility UK. In particular, the receiver or transceiver bind must use the address range parameter specified in the supplied configuration. Alternatively, when an MSISDN range has been supplied the address range parameter may be any value or range of values within the allocated MSISDN range.

1.3. Message submission

Both 7-bit and 8-bit messages may be submitted for delivery as an MT SMS to a mobile device on any supported network. The source address of the MT SMS must be any single address within the configured *address range* of your account.

1.4. Receiving incoming messages

A message sent by a mobile on a supported network to any number within the range of numbers specified in your account's *address range* will be delivered to your application. Both 7-bit and 8-bit messages are supported for mobile originated (MO) SMS. Virtual Mobile Numbers and Virtual SIM Hosting can be used to receive SMS from mobiles. More information can be found online at <https://linkmobility.co.uk/services/two-way-sms-messaging/> or to discuss please contact our [Sales](#) team.

1.5. Unbinding

Applications must conform to SMPP and issue an *unbind* PDU when they wish to disconnect. An *unbind_resp* PDU response will then be sent to the application allowing the application to then close the connection.

Periodically our systems will issue an *unbind* request to your application. An *unbind_resp* PDU should then be sent by your application, followed by the application attempting to automatically reconnect (see Re-Connect Behaviour of ESME Applications).

2. SMPP API Implementation Guidelines

2.1. Receiver Binding

When using the SMPP API receiver channel should remain permanently bound into the SMSC. If the bind fails a reconnect should be attempted (see Re-Connect Behaviour of ESME Applications).

2.2. SMSC/ESME Disconnection

The correct termination of an SMPP Session is for the ESME to issue an “unbind” request to the SMSC and vice-versa. However, it is important that the ESME is able to detect a sudden and complete closure of the underlying virtual connection, without an “unbind” message being received from the SMSC.

Enquire links can be used to detect a session that has failed. The ESME should issue an *enquire_link* following 60 seconds of session inactivity and the SMSC should respond with an *enquire_link_resp*. If the ESME doesn't receive a response from the SMSC within 60 seconds then the ESME should terminate the SMPP session.

ESMEs without the capability to detect failed SMP sessions are vulnerable to becoming inoperative until some maintenance activity is undertaken, resulting in loss of service for a period of time.

2.3. Re-Connect Behaviour of ESME Applications

This will generally be determined by the purpose of the application, however most applications that have any resiliency requirements will need to incorporate some type of re-connect algorithm in order to deal with connection/network/SMSC problems.

The ESME should:

- Attempt re-connection only in cases of a ‘no response’ situation with respect to the SMSC, loss of SMSC connection or errors received when attempting to “bind”.
- Fully close both the SMPP session and any underlying data network virtual connections at some stage in the procedure. This is advisable as the SMSC may require this in order to detect a disconnection.

The ESME **must not**:

- Continuously send “bind” requests to the SMSC without appropriate backing-off periods
- Attempt more than 10 retries in the first 60 seconds of re-connecting.
- Attempt to disconnect/re-connect as a result of received error values in the response messages (submit_sm_resp, etc) from the SMSC when in a bound state.

2.3.1. Definitions

ESME	External Short Message Entity
SMPP	Short Message Peer-to-Peer
SMSC	Short Message Service Centre