

KNOWLEDGE BASE · PUBLIC DOCUMENT

SMS delivery reports (DLR): how your application knows whether a message arrived

WhatsApp & APIs

CODE

RLX-INT-KB-6552

VERSION

v2026.10.16

COUNTRY · LANGUAGE

International · English

MozOut International

RLX Holdings (Pty) Ltd. · Group RLX Holdings

Tax no. (SARS) 943 907 6234 · Registration no. (CIPC) 2025/667953/07

<https://www.my.mozout.com/knowledgebase/6552>



The short answer (for developers): the “OK” the provider returns when you send means **accepted**, not **delivered**. Whether the message reached the phone you learn later, in a **delivery report** (DLR), and only if you asked for one: with an HTTP API it arrives as a callback to the URL you gave; with SMPP you ask with the **registered_delivery** field of `submit_sm` and the receipt comes in a **deliver_sm**. What makes it all work: storing the **message ID** returned at send time, because the report carries the same ID.

The statuses you will see

Status	What it means
Accepted (ACCEPTD, ENROUTE)	The provider or the network has the message and is trying. It has not reached the phone yet.
Delivered (DELIVRD)	The network reports that it reached the handset.
Expired (EXPIRED)	The phone was off or out of coverage until the message’s validity period ran out.
Undeliverable (UNDELIV)	The number does not exist, is barred, or the network refused it.
Rejected (REJECTD)	Refused before any attempt: sender not allowed, destination not covered by the provider, a filter.
Unknown (UNKNOWN), or no report at all	A network along the way does not return reports, or the report got lost. It does not prove failure.

Receiving and matching the reports

1

Ask for the report. With an HTTP API, give the callback URL in the request or in the provider's dashboard. With SMPP, set `registered_delivery` to request a receipt in `submit_sm`.

2

Store the ID. When the send is accepted (HTTP reply or `submit_sm_resp`), save the provider's `message_id` next to your own record, with the status "sent".

3

Handle the callback carefully: a public HTTPS URL on your site that answers quickly with 200 and does the rest afterwards. Check the signature or shared secret, if the provider uses one: **receiving a webhook in PHP and checking its signature**. If you do not answer, the provider tries again, so the same report can arrive more than once: update by ID, never insert new rows.

4

With SMPP, the receipt is a `deliver_sm` whose `esm_class` marks it as a receipt. The status is in the text (`id:... stat:DELIVRD err:...`) and, with many providers, also in the `receipted_message_id` and `message_state` fields. Always answer with `deliver_sm_resp`, or the provider sends it again.

5

Watch the ID format. Some providers return the ID in hexadecimal in `submit_sm_resp` and in decimal in the receipt (or the other way round). If nothing matches, compare both forms.

6

Decide what each final status triggers: trying another channel after “expired”, flagging the number as invalid after “undeliverable”, and a deadline after which “no report” becomes “unknown”.

“Delivered” means it reached the phone, not that it was read. And reports can arrive out of order (“delivered” before “accepted”): never let an intermediate status overwrite a final one.

For verification codes, do not wait for the report before letting the user carry on: offer “resend” after a pause, and use reports for statistics and to spot numbers or networks with problems. More in **one-time codes by SMS, safely**.

The callback URL on your site receives nothing? Tell us the URL and the time of a test send, and we will check whether the request reached the server.

[Open a support ticket](#)

SEE ALSO

SMPP or an HTTP SMS API: what each one is and which your application needs

SMPP binds and enquire_link: keeping an SMS connection alive on a VPS

Receiving a webhook in PHP and checking its signature

What a webhook is, and how to test one with curl

Webhook or polling: how one system learns that something changed in another

Sending one-time codes (OTP) by SMS safely: expiry, attempt limits and logs



Online version of this document

MozOut is a leading internet services company, with a mission to help businesses to be born and prosper through technology.

2020

FOUNDED

37,000+

CLIENTS

130+

SIGN-UPS A DAY

MozOut International

RLX Holdings (Pty) Ltd. • Group RLX Holdings

TAX NO. (SARS)

943 907 6234

REGISTRATION NO.
(CIPC)

2025/667953/07

REGISTERED OFFICE

1111 Olive Street
Saint Louis 63101, Missouri, United States

CONTACT

info@mozout.com • +27 78 823 3325 • www.mozout.com

OPENING HOURS

Monday to Friday - 08 AM to 04 PM
Saturday & Sunday - Closed