



GUIDES · API V1

Rate limits

Limits keep delivery healthy for everyone and stop OTP abuse. Every limit answers 429 `rate_limited`, with `Retry-After` whenever it is known.

Updated 2026-10-10 · generated 2026-10-10 · smsray.com/docs/rate-limits

Per-client rate

Each API client may make up to **20 requests per second** by default, across all v1 endpoints. The limit is per client, so separate apps on separate clients don't slow each other down. If you need more, talk to us.

SENDING SPEED

The request rate is how fast we accept messages. Delivery is paced by our platform to protect deliverability, so a large burst is accepted at once and delivered over the following minutes.

OTP limits

POST `/sms/otp/send` has extra limits that protect your users and your balance from SMS pumping and brute force:

LIMIT	SCOPE	VALUE
Resend cooldown	Your client + number + purpose	1 code per 60 s
Per number	The phone number, across every SMSRay customer	3 codes per 10 min
Per end user IP	Your client + the <code>ip</code> you pass	10 codes per 10 min
Per calling IP	Your client + your server's IP	60 requests per minute
Verify attempts	Each code	5 wrong tries, then locked
Code lifetime	Each code	5 minutes

The per-IP limit applies only when you pass `ip`. Pass your end user's IP, not your server's.

The 429 response

Every limit answers the same way. `Retry-After` (in seconds) is set whenever the wait is known, including the OTP resend cooldown.

429 response

```
HTTP/1.1 429 Too Many Requests
Retry-After: 42
content-type: application/json
```

```
{ "error": "Please wait before requesting another code", "code": "rate_limited" }
```

Retrying well

- Honour Retry-After when present; otherwise back off exponentially with jitter.
- For OTP, show the remaining cooldown in your UI instead of letting users hammer the resend button.
- Reuse the same Idempotency-Key when retrying a send so a retry can never double-send. See [Idempotency](#).
- Smooth large batches on your side to stay under your per-second rate.
- Optional per-key caps (daily cap, per-number hourly cap) answer 429 `daily_cap` or `number_cap`. See [Errors](#).

Webhook retries

Limits run the other way too: when your webhook endpoint fails or is slow, SMSRay retries up to 8 times over about 4.5 hours (10 s, 30 s, 2 min, 10 min, 30 min, 1 h, 3 h), and stops after 3 attempts on a permanent 4xx. See [Retries and timeouts](#).

The web page is always the current version: smsray.com/docs/rate-limits

Full API reference PDF: smsray.com/docs/pdf · Your keys and live examples: app.smsray.com/developers