

MHI Gateway

Rate Limits Guide

Request throttling and durable outbound SMS usage limits

Document version 1.0

Publication date 10 August 2026

Classification Public integration documentation

MICRO HEALTH INITIATIVE

Contents

Use this contents list or the PDF bookmark panel to navigate.

Contents	2
1. Two independent controls	3
2. Request throttling	3
3. Durable outbound SMS quotas	3
4. Usage units	3
5. Acceptance and rejection	4
Single-message request	4
Bulk request	4
Idempotent replay	4
6. USAGE_LIMIT_EXCEEDED	4
7. Developer response	4
8. Integration checklist	4
9. Source and scope	5

1. Two independent controls

MHI Gateway uses two distinct controls that may both return HTTP 429. Integrations must branch on the public error code.

Control	Code	Purpose
Request throttling	RATE_LIMIT_EXCEEDED	Controls request frequency and protects service capacity.
Durable SMS quota	USAGE_LIMIT_EXCEEDED	Controls accepted outbound SMS usage for tenant and application scopes.

2. Request throttling

Request throttling concerns how frequently the API is called. When it applies, the gateway returns 429 `RATE_LIMIT_EXCEEDED`.

- Honor `Retry-After` when it is present.
- Use bounded exponential backoff with jitter.
- Preserve the original request and `Idempotency-Key` for safe submission retries.
- Do not infer durable quota state from a throttling response.

3. Durable outbound SMS quotas

Durable quotas apply to accepted outbound SMS usage. Limits may be configured independently at tenant and application scope for daily and monthly UTC periods. Every applicable scope and period must permit the prospective usage.

Dimension	Approved behavior
Tenant	Aggregates outbound SMS usage across its applications.
Application	Applies only to the authenticated application.
Daily period	UTC calendar day.
Monthly period	UTC calendar month.
Unlimited	A null limit means no configured quota for that scope and period.
Zero	A zero limit blocks outbound SMS acceptance.

4. Usage units

Usage is measured using the authoritative `total_parts` produced by SMS segmentation. A multipart message therefore consumes one unit per persisted SMS part, not merely one unit per API item.

Exactly one durable `outbound_sms` usage record is created per accepted outbound message. Idempotent replay, outbox replay, SMPP retries, repeated submission attempts, delivery receipts, and final-status changes do not create additional usage.

5. Acceptance and rejection

Single-message request

Before accepting a message, the gateway evaluates its prospective SMS parts against the applicable daily and monthly tenant and application limits. If any configured limit would be exceeded, the request is rejected before message, SMS, event, outbox, usage, or counter persistence.

Bulk request

Bulk processing is normally best-effort after top-level validation. Quota enforcement is a narrower request-level rule: the complete bulk request's prospective accepted units are checked before item persistence. If the quota would be exceeded, the whole request returns `USAGE_LIMIT_EXCEEDED` with no partial message or usage persistence.

Idempotent replay

An equivalent replay returns the original stable result and does not consume usage again. A changed payload under the same key returns `IDEMPOTENCY_CONFLICT`.

6. USAGE_LIMIT_EXCEEDED

```
HTTP/1.1 429 Too Many Requests
Content-Type: application/json

{
  "error": {
    "code": "USAGE_LIMIT_EXCEEDED",
    "message": "The outbound SMS usage limit has been exceeded.",
    "details": {},
    "correlation_id": "550e8400-e29b-41d4-a716-446655440000"
  }
}
```

The response intentionally does not reveal current usage, configured values, another application's totals, the exact failed scope, wallet state, or pricing information.

7. Developer response

Observed code	Recommended action
RATE_LIMIT_EXCEEDED	Pause according to Retry-After when present, then retry with bounded backoff.
USAGE_LIMIT_EXCEEDED	Stop automatic throttling retries. Contact the tenant administrator or wait for the applicable UTC period to change.

- Persist the original Idempotency-Key before the first submission attempt.
- Track message parts when estimating usage, while treating the gateway's authoritative `total_parts` as final.
- Design bulk clients for request-level quota rejection before any item is accepted.
- Never attempt to discover another tenant or application's quota state through errors.

8. Integration checklist

- Differentiate the two 429 codes in application logic and monitoring.
- Use UTC boundaries for operational expectations around daily and monthly limits.
- Preserve idempotency during safe retries.

- Treat null as unlimited only where the operator has configured it; zero blocks usage.
- Escalate durable quota errors to the tenant administrator without logging message content or credentials.

9. Source and scope

This guide is derived from the approved public API contract, Developer Integration Guide, and Usage & Limits architecture decision. It documents existing public behavior only and intentionally omits internal counter implementation and commercial features.