

MHI Gateway

Authentication Guide

Bearer credentials, request context, idempotency, and secure integration

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. Authentication at a glance	3
2. Credentials	3
Issuance and application association	3
Expiry, revocation, and rotation	3
Secure handling	3
3. Authenticated request example	3
4. Request identifiers	3
X-Correlation-ID	3
Response identifiers	4
5. Idempotency-Key	4
6. Authorization and isolation	4
7. Authentication failures	4
8. Production checklist	5
9. Source and scope	5

1. Authentication at a glance

Every public API operation uses HTTP Bearer authentication. A credential resolves to one active tenant and application; all resource access remains within that authenticated scope.

```
Authorization: Bearer <API_TOKEN>
Accept: application/json
Content-Type: application/json
```

Treat the complete credential as opaque. Do not parse its prefix or secret to make authorization, routing, or environment decisions.

2. Credentials

Issuance and application association

Obtain API credentials from the gateway operator. Issuance, expiry, revocation, and storage are operator-managed. Each credential is associated with a tenant and application, and cannot be used to cross those boundaries.

Expiry, revocation, and rotation

An expired, revoked, missing, or invalid credential cannot authenticate a request. Stop using a credential immediately after revocation. Coordinate rotation with the gateway operator; the public contract does not promise an undocumented overlap period.

Secure handling

- Store credentials in a managed secret store and inject them only at runtime.
- Use separate credentials for each environment and workload.
- Never commit credentials or expose the Authorization header in logs, traces, errors, or screenshots.
- Use HTTPS and verify server certificates for production requests.

3. Authenticated request example

```
curl --request POST "${BASE_URL}/api/v1/messages" \
  --header "Authorization: Bearer ${API_TOKEN}" \
  --header "Accept: application/json" \
  --header "Content-Type: application/json" \
  --header "Idempotency-Key: 6b11f086-1e8d-4bb9-a680-1524e8d23b21" \
  --header "X-Correlation-ID: order-1042-attempt-1" \
  --data '{
    "recipient": "+255700000000",
    "sender_id": "MHI",
    "message": "Hello from MHI Gateway"
  }'
```

Set `BASE_URL` to the operator-provided deployment URL and load `API_TOKEN` from a secret manager. Do not hard-code either value in source code.

4. Request identifiers

X-Correlation-ID

Clients may supply X-Correlation-ID to connect client and gateway telemetry. It must contain 1-100 characters, begin with an ASCII letter or digit, and then use only letters, digits, period, underscore, colon, or hyphen.

Response identifiers

Header	Meaning
X-Request-ID	Unique identifier for the current HTTP request.
X-Correlation-ID	The valid client value, or a gateway-generated UUIDv4, for the current request.

Capture both headers on success and failure. On idempotent replay, response headers identify the current HTTP exchange while a single-message body correlation_id retains the original submission identity.

5. Idempotency-Key

Both message submission operations require Idempotency-Key. Use visible ASCII, 1-191 characters, and generate a fresh unpredictable value for each intended operation.

- Equivalent replay returns the original result and status code.
- Changed canonical input under the same key returns 409 IDEMPOTENCY_CONFLICT.
- Failed validation does not bind the key.
- Concurrent equivalent requests converge on one stable result.
- Bulk array order is contract-significant.

After an ambiguous timeout, retry the identical request using the original key. Do not silently replace the key.

6. Authorization and isolation

Credentials are tenant-scoped and application-associated. A credential cannot access another tenant's messages. Permission failures may return 403 FORBIDDEN. A resource outside the authenticated scope is reported as absent so error behavior cannot reveal another tenant's data.

7. Authentication failures

HTTP	Code	Meaning	Developer response
401	AUTHENTICATION_FAILED	Credential is absent, invalid, expired, or revoked.	Supply an active credential; do not retry an unchanged invalid credential.
403	FORBIDDEN	The authenticated principal lacks permission.	Request appropriate access or use the correct credential.
422	INVALID_REQUEST	A request header or payload is invalid.	Correct the request using safe validation details.
409	IDEMPOTENCY_CONFLICT	The key is bound to different canonical input.	Restore the original request or create a new operation with a new key.

```
{
  "error": {
    "code": "AUTHENTICATION_FAILED",
    "message": "Authentication failed.",
    "details": {},
    "correlation_id": "550e8400-e29b-41d4-a716-446655440000"
  }
}
```

Branch on the stable error code, not the message. Never include the credential in a support ticket; provide safe request identifiers instead.

8. Production checklist

- Use HTTPS and certificate validation.
- Load credentials from a secret manager and redact Authorization.
- Persist idempotency keys with requests and stable results.
- Capture X-Request-ID and X-Correlation-ID on every response.
- Use bounded timeouts and retry only when the operation is safe and idempotent.
- Verify tenant and application context before processing returned resources.

9. Source and scope

This guide is derived from the approved Developer Integration Guide and public API contract. It documents the public integration surface only; operator implementation details and secrets are intentionally excluded.