
C_12509_Anlage - Push Notifications - Push Gateway API - Response Status 422 wird entfernt und Datei umstrukturiert

Inhaltsverzeichnis

1 Änderungsbeschreibung.....	2
2 Änderung in push_gateway_openapi.yaml.....	3

1 Änderungsbeschreibung

Der HTTP-Status 422 wird beim Push Gateway verwendet, um zu melden, dass ein Batch nur teilweise bearbeitet werden kann. Der HTTP-Status 200 deckt diese Situation bereits über das status Feld im Ergebnis mit den Werte success, partial, und failed ab. Dieser Änderungseintrag entfernt den HTTP-Status 422 für das Push Gateway. Darüber hinaus wird die Datei umstrukturiert, um sie besser für Code-Generatoren zu gestalten.

2 Änderung in push_gateway_openapi.yaml

Folgendes wird aus der Datei push_gateway_openapi.yaml entfernt:

```
'422':
description: Batch partially processed - some notifications failed validation
content:
  application/json:
    schema:
      $ref: '#/components/schemas/BatchNotificationResponse'
    examples:
      partialFailure:
        value:
          results:
            - id: "batch_item_1"
              status: "success"
              rejected: []
            - id: "batch_item_2"
              status: "failed"
              error: "Invalid notification format"
          summary:
            total: 2
            successful: 1
            failed: 1
            partial: 0
```

Folgende Beschreibungen haben sich geändert:

Stelle	Alt	Neu
EncryptedNotification.identifier	Random identifier to reidentify the push notification at the originating fachdienst. This property may be send to a 3rd service (e.g. in combination with some properties from the `data` dictionary) to create a complete list of all push notifications that have been sent to a client. This property is optional and must only be set by a fachdienst, if the fachdienst implements the history endpoints.	An identifier to reidentify the push notification at the originating fachdienst. This property may be send to a 3rd service (e.g. in combination with some properties from the `data` dictionary) to create a complete list of all push notifications that have been sent to a client. This property is optional and must only be set by a fachdienst, if the fachdienst implements the history endpoints. NOTE: It is possible that this identifier is not unique across multiple push notifications, e.g. if the fachdienst sends multiple push notifications for the same

		event (e.g. in a retry scenario or multiple push notifications being sent for one event).
POST /notify.description	-	Submit a plain push notification to the Push Gateway. The notification will be sent to all devices listed in the `devices` array of the notification object. *This mimics the Matrix Push Gateway API in version 1.0.0.*
/notify/batch /notifyEncrypted/batch Response 200	Batch processing results with per-item status and summary.	Batch processing results with per-item status and summary. If the Push Gateway is unable to complete processing for some notifications, those will be marked as failed in the results.

Die Datei wurde umstrukturiert und sieht jetzt wie folgt aus:

Link: https://github.com/gematik/gem-push-notifications-concept/blob/1.0.1/docs_sources/push_gateway_openapi.yaml

```
# Copyright 2024 Gematik GmbH
#
# Based on the OpenAPI specification of the Matrix Client-Server API
https://github.com/matrix-org/matrix-spec
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#   http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
openapi: 3.1.0
info:
  title: Push Gateway API
  description: API for interacting with the Push Gateway
  version: 1.0.0
  license:
    name: Apache 2.0
    url: http://www.apache.org/licenses/LICENSE-2.0.html
servers:
  - url: "{protocol}://{hostname}{basePath}"
```

```

variables:
  protocol:
    enum:
      - https
    default: https
  hostname:
    default: localhost:8080
  basePath:
    default: /push/v1
paths:
/notify:
  post:
    operationId: push_v1_notify_plain
    summary: Push plain data to the gateway
    description: |-
      Submit a plain push notification to the Push Gateway. The notification
      will be sent to all devices listed in the `devices` array of the
      notification object.

```

This mimics the Matrix Push Gateway API in version 1.0.0.

```

requestBody:
  required: true
  content:
    application/json:
      examples:
        plainPushNotification:
          value:
            {
              "notification": {
                "event_id": "event_id",
                "room_id": "room_id",
                "type": "type",
                "sender": "sender",
                "sender_display_name": "sender_display_name",
                "room_name": "room_name",
                "room_alias": "room_alias",
                "user_is_target": true,
                "prio": "high",
                "content": {
                  "content": "content"
                },
              },
              "counts": {
                "unread": 1,
                "missed_calls": 0
              },
              "devices": [
                {
                  "app_id": "app_id",
                  "pushkey": "pushkey",
                  "pushkey_ts": 1634025600,
                  "data": {
                    "format": "format"
                  },
                },
                {
                  "tweaks": {
                    "tweak": "tweak"
                  }
                }
              ]
            }

```

```

    }
  ]
}
}
schema:
  $ref: '#/components/schemas/PlainNotification'
responses:
  "200":
    description: A list of rejected push keys.
    content:
      application/json:
        schema:
          type: object# empty json object
          properties:
            rejected:
              type: array
              description: |-
                A list of all pushkeys given in the notification request that
                are not valid. These could have been rejected by an upstream
                gateway because they have expired or have never been valid.
                Homeservers must cease sending notification requests for these
                pushkeys and remove the associated pushers. It may not
                necessarily be the notification in the request that failed:
                it could be that a previous notification to the same pushkey
                failed. May be empty.
              items:
                type: string
                description: A pushkey that has been rejected.
              required:
                - rejected
            examples:
              response:
                value: {
                  "rejected": [
                    "V2h5lG9ulGVhcnRoIGRpZCB5b3UgZGVjb2RlIHRobaXM/"
                  ]
                }
  '400':
    description: Invalid input
    content:
      application/json:
        schema:
          type: object
          properties:
            error:
              type: string
              example: "Invalid data format"
  '401':
    description: The client is not authenticated.
    content:
      application/json:
        schema:
          type: object
          properties:
            error:
              type: string

```

```
    example: "Missing or invalid mutual TLS (mTLS) certificate."
'413':
  description: The request payload exceeds the maximum allowed size.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Notification object is too large."
'500':
  description: A generic error occurred on the Push Gateway.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Unexpected failure during processing."
'503':
  description: The Push Gateway is temporarily unavailable.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Push Gateway under maintenance or overloaded."
security:
  - mtls: []
tags:
  - Push Gateway
/notify/batch:
  post:
    operationId: push_v1_notify_batch_plain
    summary: Push multiple plain notifications to the gateway in a single request
    description: |-
      Submit multiple plain push notifications in a single batch request. This endpoint
      is optimized for high-throughput scenarios where multiple plain notifications
      need to be sent efficiently. The batch can contain up to a maximum of 100 notifications
      per request.
    requestBody:
      required: true
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/BatchNotificationRequest'
          examples:
            plainBatch:
              value:
                notifications:
                  - id: "batch_item_1"
                    notification:
```

```

event_id: "event_id_1"
room_id: "room_id_1"
type: "m.room.message"
sender: "sender_1"
sender_display_name: "Sender Display Name"
room_name: "Room Name"
prio: "high"
content:
  content: "Message content"
counts:
  unread: 1
  missed_calls: 0
devices:
- app_id: "app_id_1"
  pushkey: "pushkey_1"
  pushkey_ts: 1634025600
  data:
    format: "format"
    tweaks:
      tweak: "tweak"
- id: "batch_item_2"
notification:
  event_id: "event_id_2"
  room_id: "room_id_2"
  type: "m.room.message"
  sender: "sender_2"
  sender_display_name: "Sender Display Name"
  room_name: "Room Name"
  prio: "high"
  content:
    content: "Message content"
  counts:
    unread: 2
    missed_calls: 0
  devices:
  - app_id: "app_id_2"
    pushkey: "pushkey_2"
    pushkey_ts: 1634025600
    data:
      format: "format"
    tweaks:
      tweak: "tweak"

```

responses:

"200":

description: Batch processing results with per-item status and summary. If the Push Gateway is unable to complete processing for some notifications, those will be marked as failed in the results.

content:

application/json:

schema:

\$ref: '#/components/schemas/BatchNotificationResponse'

examples:

batchResponse:

value:

results:

- id: "batch_item_1"


```
    status: "success"
    rejected: []
  - id: "batch_item_2"
    status: "partial"
    rejected: ["expired_pushkey_123"]
summary:
  total: 2
  successful: 1
  failed: 0
  partial: 1
partialFailure:
value:
  results:
    - id: "batch_item_1"
      status: "success"
      rejected: []
    - id: "batch_item_2"
      status: "failed"
      error: "Invalid notification format"
summary:
  total: 2
  successful: 1
  failed: 1
  partial: 0
'400':
description: Invalid batch request
content:
  application/json:
    schema:
      type: object
      properties:
        error:
          type: string
          enum:
            - "Invalid data format"
            - "Batch size exceeds maximum limit"
            - "Duplicate notification IDs in batch"
            - "Individual notification too large"
      example: "Batch size exceeds maximum limit"
details:
  type: object
  properties:
    max_batch_size:
      type: integer
      example: 100
    max_notification_size_kb:
      type: integer
      example: 64
    duplicate_ids:
      type: array
      items:
        type: string
      example: ["batch_item_1", "batch_item_1"]
    current_batch_size:
      type: integer
      example: 150
```

```
'401':
  description: The client is not authenticated.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Missing or invalid mutual TLS (mTLS) certificate."
'413':
  description: The request payload exceeds the maximum allowed size.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Batch request payload is too large."
'500':
  description: A generic error occurred on the Push Gateway.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Unexpected failure during batch processing."
'503':
  description: The Push Gateway is temporarily unavailable.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Push Gateway under maintenance or overloaded."
security:
  - mtls: []
tags:
  - Push Gateway
/notifyEncrypted/batch:
  post:
    operationId: push_v1_notify_batch_encrypted
    summary: Push multiple encrypted notifications to the gateway in a single request
    description: |-
      Submit multiple encrypted push notifications in a single batch request. This endpoint
      is optimized for high-throughput scenarios where multiple encrypted notifications
      need to be sent efficiently. The batch can contain up to a maximum of 100 notifications
      per request.
    requestBody:
      required: true
      content:
```

```

application/json:
  schema:
    $ref: '#/components/schemas/EncryptedBatchNotificationRequest'
  examples:
    encryptedBatch:
      value:
        notifications:
          - id: "enc_batch_1"
            notification:
              ciphertext: "base64_of_ciphertext_1"
              time_message_encrypted: "2024-11"
              key_identifier: "123e4567-e89b-12d3-a456-426614174000"
              prio: "high"
              counts:
                unread: 1
                missed_calls: 0
              device:
                app_id: "app_id"
                pushkey: "pushkey"
                pushkey_ts: 1634025600
              data:
                format: "format"
              tweaks:
                tweak: "tweak"
          - id: "enc_batch_2"
            notification:
              ciphertext: "base64_of_ciphertext_2"
              time_message_encrypted: "2024-11"
              key_identifier: "456e7890-e89b-12d3-a456-426614174001"
              prio: "low"
              counts:
                unread: 2
                missed_calls: 0
              device:
                app_id: "app_id"
                pushkey: "pushkey"
                pushkey_ts: 1634025600
              data:
                format: "format"
              tweaks:
                tweak: "tweak"

```

responses:

"200":

description: Batch processing results with per-item status and summary. If the Push Gateway is unable to complete processing for some notifications, those will be marked as failed in the results.

content:

```

application/json:
  schema:
    $ref: '#/components/schemas/BatchNotificationResponse'
  examples:
    batchResponse:
      value:
        results:
          - id: "enc_batch_1"
            status: "success"

```

```

    rejected: []
    - id: "enc_batch_2"
      status: "partial"
      rejected: ["expired_pushkey_123"]
  summary:
    total: 2
    successful: 1
    failed: 0
    partial: 1
  partialFailure:
  value:
    results:
      - id: "enc_batch_1"
        status: "success"
        rejected: []
      - id: "enc_batch_2"
        status: "failed"
        error: "Invalid notification format"
    summary:
      total: 2
      successful: 1
      failed: 1
      partial: 0
'400':
  description: Invalid batch request
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            enum:
              - "Invalid data format"
              - "Batch size exceeds maximum limit"
              - "Duplicate notification IDs in batch"
              - "Individual notification too large"
          example: "Batch size exceeds maximum limit"
      details:
        type: object
        properties:
          max_batch_size:
            type: integer
            example: 100
          max_notification_size_kb:
            type: integer
            example: 64
          duplicate_ids:
            type: array
            items:
              type: string
            example: ["enc_batch_1", "enc_batch_1"]
          current_batch_size:
            type: integer
            example: 150
'401':

```

```
description: The client is not authenticated.
content:
  application/json:
    schema:
      type: object
      properties:
        error:
          type: string
          example: "Missing or invalid mutual TLS (mTLS) certificate."
'413':
  description: The request payload exceeds the maximum allowed size.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Batch request payload is too large."
'500':
  description: A generic error occurred on the Push Gateway.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Unexpected failure during batch processing."
'503':
  description: The Push Gateway is temporarily unavailable.
  content:
    application/json:
      schema:
        type: object
        properties:
          error:
            type: string
            example: "Push Gateway under maintenance or overloaded."
security:
  - mtls: []
tags:
  - Push Gateway
components:
securitySchemes:
  mtls:
    description: |-
      Mutual TLS authentication. The server will verify the client's certificate and reject the
      connection if it is invalid.
    type: mutualTLS
schemas:
  EncryptedNotification:
    type: object
    title: EncryptedNotification
    description: Information about the push notification
    required:
```

- device
- ciphertext
- time_message_encrypted
- key_identifier

properties:

ciphertext:

- type: string
- description: "Base64 encoded ciphertext. The original ciphertext always contains 1024 bytes of payload. Encoding this using Base64 results in 1368 bytes for this data field."

time_message_encrypted:

- type: string
- description: "Year + Month the key was derived and the message was sent. e.g. `2024-11` for November 2024."

key_identifier:

- type: string
- description: |-

A string containing an identifier for the key. This is used to identify the key and the corresponding service (Fachdienst) when a push notification is received within the FdV. The `key_identifier` is generated by the FdV, must be random and must not be reused for a different ISS. An UUID is recommended.

identifier:

- type: string
- description: |-

An identifier to reidentify the push notification at the originating fachdienst. This property may be send to a 3rd service (e.g. in combination with some properties from the `data` dictionary) to create a complete list of all push notifications that have been sent to a client. This property is optional and must only be set by a fachdienst, if the fachdienst implements the history endpoints.

NOTE: It is possible that this identifier is not unique across multiple push notifications, e.g. if the fachdienst sends multiple push notifications for the same event (e.g. in a retry scenario or multiple push notifications being sent for one event).

prio:

- \$ref: "#/components/schemas/Priority"

counts:

- \$ref: "#/components/schemas/Counts"

device:

- \$ref: "#/components/schemas/Device"

PlainNotification:

- type: object
- title: PlainNotification
- description: Information about the push notification
- required:
 - devices
 - prio
- properties:
 - event_id:
 - type: string
 - description: |-

The Matrix event ID of the event being notified about. This is required if the notification is about a

particular Matrix event. It may be omitted for notifications that only contain updated badge counts. This ID can and should be used to detect duplicate notification requests.

room_id:

type: string

description: |-

The ID of the room in which this event occurred.

Required if the notification relates to a specific

Matrix event.

type:

type: string

description: The type of the event as in the event's `type` field.

sender:

type: string

description: The sender of the event as in the corresponding event field.

sender_display_name:

type: string

description: |-

The current display name of the sender in the room in which the event occurred.

room_name:

type: string

description: The name of the room in which the event occurred.

room_alias:

type: string

description: An alias to display for the room in which the event occurred.

user_is_target:

type: boolean

description: |-

This is true if the user receiving the notification is the subject of a member event (i.e. the `state_key` of the member event is equal to the user's Matrix ID).

prio:

\$ref: "#/components/schemas/Priority"

content:

type: object

title: EventContent

description: |-

The `content` field from the event, if present. The pusher may omit this if the event had no content or for any other reason.

counts:

\$ref: "#/components/schemas/Counts"

devices:

type: array

title: Devices

description: This is an array of devices that the notification should be sent to.

items:

\$ref: "#/components/schemas/Device"

Counts:

type: object

title: Counts

description: |-

This is a dictionary of the current number of unacknowledged communications for the recipient user. Counts whose value is zero should be omitted.

```
properties:
  unread:
    type: integer
    description: |-
      The number of unread messages a user has across all of the
      rooms they are a member of.
  missed_calls:
    type: integer
    description: |-
      The number of unacknowledged missed calls a user has
      across all rooms of which they are a member.
BatchNotificationRequest:
  type: object
  properties:
    notifications:
      type: array
      minItems: 1
      maxItems: 100
      description: Array of plain notifications to process in this batch
    items:
      type: object
      properties:
        id:
          type: string
          maxLength: 64
          description: Client-provided identifier for tracking this notification in the
response
  notification:
    $ref: '#/components/schemas/PlainNotification'
  required:
    - id
    - notification
  required:
    - notifications
EncryptedBatchNotificationRequest:
  type: object
  properties:
    notifications:
      type: array
      minItems: 1
      maxItems: 100
      description: Array of encrypted notifications to process in this batch
    items:
      type: object
      properties:
        id:
          type: string
          maxLength: 64
          description: Client-provided identifier for tracking this notification in the
response
  notification:
    $ref: '#/components/schemas/EncryptedNotification'
  required:
    - id
    - notification
  required:
```


- notifications

BatchNotificationResponse:

type: object

properties:

results:

type: array

description: Results for each notification in the batch, in the same order as submitted

items:

type: object

properties:

id:

type: string

description: The client-provided identifier from the request

status:

type: string

enum: [success, failed, partial]

description: |-

Processing status for this notification:

- `success`: The notification was processed successfully and will be sent to all target devices. All pushkeys were valid and accepted by the upstream gateway for past notifications (if any).
- `failed`: The notification failed to process completely. This could be due to invalid notification format, missing required fields, or a complete failure in reaching any target devices. Check the `error` field for details.
- `partial`: The notification was partially successful. Can only happen if multiple devices were to receive the same message (using `notify` or `notify/batch`). Some pushkeys were accepted and the notification will be sent to those devices, but other pushkeys were rejected (listed in the `rejected` array). This typically happens when some pushkeys have expired or become invalid (e.g. on device restore or app reinstallation).

rejected:

type: array

items:

type: string

description: List of rejected pushkeys for this notification (when status is success or partial)

error:

type: string

description: Error message when status is failed

required:

- id
- status

summary:

type: object

properties:

total:

type: integer

description: Total number of notifications in the batch

successful:

type: integer

description: Number of successfully processed notifications

failed:

type: integer

description: Number of failed notifications

partial:

type: integer

description: Number of partially successful notifications (some pushkeys rejected)

required:

- total
- successful
- failed
- partial

required:

- results
- summary

Priority:

type: string

enum:

- high
- low

default: high

description: |-
The priority of the notification. If omitted, `high` is assumed. This may be used by push gateways to deliver less time-sensitive notifications in a way that will preserve battery power on mobile devices.

Device:

type: object

title: Device

description: The device that the notification should be sent to.

properties:

app_id:

type: string

description: The `app_id` given when the pusher was created.

pushkey:

type: string

description: The `pushkey` given when the pusher was created.

pushkey_ts:

type: integer

format: int64

description: |-
The unix timestamp (in seconds) when the pushkey was last updated.

data:

type: object

title: PusherData

description: |-
A dictionary of additional pusher-specific data. This is the `data` dictionary passed in at [pusher creation](/client-server-api/#post_matrixclientv3pushersset) minus the `url` key.

properties:

format:

type: string

description: |-
The format to use for sending notifications.

tweaks:

type: object

title: Tweaks

description: |-
A dictionary of customisations made to the way this notification is to be presented. These are added by push rules.

required:

- app_id
- pushkey