HIS 醫令與健檢推送測試指引

- 文件版本：`v1.0`
- 更新日期：`2026-06-23`
- 適用對象：HIS 廠商測試端、整合端
- 目的：提供可直接照做的 `curl` / 命令列範例，方便驗證醫令推送與健檢同步流程

1. 先備條件

你已經拿到：

- HIS 登入帳號 / 密碼
- `HIS_VENDOR_CLIENT_SECRET`
- 後續會另外提供的 AES 加密 key

請先參考已提供的 `login_api.pdf` 取得 `access_token`。後續兩個 API 都需要帶入 `Authorization: Bearer <access_token>`。

建議先設定環境變數

```
export HIS_TEST_BASE_URL='https://healthcheck-backend-ts.delixir.cc'
export HIS_VENDOR_ACCOUNT='你的帳號'
export HIS_VENDOR_PASSWORD='你的密碼'
export HIS_VENDOR_CLIENT_SECRET='你的secret'
export HIS_AES_KEY_BASE64='你的AES key base64'
```

> AES key 尚未提供前，可以先把文件流程看熟；等 key 拿到後再執行加密與送出。

2. 共通流程

兩個推送 API 的流程都相同：

1. 先登入取得 `access_token`
2. 準備明文 JSON payload
3. 用 AES-256-GCM 將 payload 加密，包成 `encryption + data` 的 envelope
4. 用 `curl` 送出 envelope JSON
5. 伺服器回傳處理結果

> `base64` 只是編碼，不是加密。真正的保護是 AES-256-GCM。

2.1 加密範例

下面範例都假設 `payload` 已經在記憶體中，函式會回傳可直接送出的 envelope 物件。

規則很固定：

- `algorithm` 固定為 `AES-256-GCM`
- `key_id` 直接用 `$HIS_VENDOR_ACCOUNT`

- `iv` 用隨機 12 bytes，再 Base64
- `tag` 用 16 bytes auth tag，再 Base64
- `data` 放 ciphertext 的 Base64

> Python 範例需要 `cryptography`。

Python

```
import base64, json
from secrets import token_bytes
from cryptography.hazmat.primitives.ciphers.aead import AESGCM

def encrypt(payload, key_b64, vendor_account):
    key = base64.b64decode(key_b64)
    iv = token_bytes(12)
    ct = AESGCM(key).encrypt(iv, json.dumps(payload, separators=(',', ':'),
ensure_ascii=False).encode(), None)
    return {"encryption": {"algorithm": "AES-256-GCM", "key_id":
vendor_account, "iv": base64.b64encode(iv).decode(), "tag":
base64.b64encode(ct[-16:]).decode(), "encoding": "base64"}, "data":
base64.b64encode(ct[:-16]).decode()}
```

Node.js

```
const crypto = require('crypto')
const encrypt = (payload, keyB64, vendorAccount) => { const iv =
crypto.randomBytes(12), c = crypto.createCipheriv('aes-256-gcm',
Buffer.from(keyB64, 'base64'), iv), ct =
Buffer.concat([c.update(JSON.stringify(payload)), c.final()]); return {
encryption: { algorithm: 'AES-256-GCM', key_id: vendorAccount, iv:
iv.toString('base64'), tag: c.getAuthTag().toString('base64'), encoding:
'base64' }, data: ct.toString('base64') } }
```

C#

```
using System.Security.Cryptography;
using System.Text;
using System.Text.Json;
static object Encrypt(object payload, string keyB64, string vendorAccount)
{ var key = Convert.FromBase64String(keyB64); var iv =
RandomNumberGenerator.GetBytes(12); var pt =
Encoding.UTF8.GetBytes(JsonSerializer.Serialize(payload)); var ct = new
byte[pt.Length]; var tag = new byte[16]; using var aes = new AesGcm(key);
aes.Encrypt(iv, pt, ct, tag); return new { encryption = new { algorithm =
"AES-256-GCM", key_id = vendorAccount, iv = Convert.ToBase64String(iv), tag
```

```
= Convert.ToBase64String(tag), encoding = "base64" }, data =
Convert.ToBase64String(ct) }; }
```

如果要直接送出，最簡單就是把上面程式產生的 JSON 字串放進 `BODY`，再交給 `curl --data-raw "$BODY"`。

2.2 `curl` 送出時要帶的共通 Header

兩個 API 都需要：

- `Authorization: Bearer <access_token>`
- `Content-Type: application/json`
- `X-Timestamp`
- `X-Nonce`
- `X-Signature`
- `X-Request-Id`

各欄位取得方式如下：

Header	取得方式	補充說明
`X-Timestamp`	使用送出當下的 Unix Timestamp（秒）	例如 `1719123456`。不可用 ISO 8601。
`X-Nonce`	每次 request 產生一個新的隨機字串	建議至少 16 bytes 以上亂數來源。不可重複使用，否則會被視為 replay request。
`X-Signature`	用 `client_secret` 對 canonical string 做 HMAC-SHA256，再將結果 Base64 編碼	必須在 request body 完全定稿後計算，因為簽章內容包含 `SHA256(Raw Request Body)`。
`X-Request-Id`	每次 request 產生一個新的唯一識別碼	建議直接使用 UUID v4，主要用於追蹤與除錯。

實務上建議流程：

1. 先完成加密後的 envelope JSON。
2. 將最終送出的 body 序列化成字串或 bytes。
3. 產生 `X-Timestamp`、`X-Nonce`、`X-Request-Id`。
4. 用同一份 body 計算 `X-Signature`。
5. 帶著這組 header 送出 request。

`X-Signature` 的計算方式如下：

```
canonical_string = X-Timestamp + "\\n" + X-Nonce + "\\n" + HTTP Method +
"\\n" + Path[?QueryString] + "\\n" + SHA256(Raw Request Body)
X-Signature = Base64(HMAC-SHA256(client_secret, canonical_string))
```

3. 醫令推送 API

3.1 端點

- Method：`POST`
- Path：`/api/v1/his/medical-order/push`

3.2 Request Body

送出的 body 必須是加密後的 envelope：

```
{
  "encryption": {
    "algorithm": "AES-256-GCM",
    "key_id": "$HIS_VENDOR_ACCOUNT",
    "iv": "Base64 編碼後之 IV / Nonce",
    "tag": "Base64 編碼後之 Authentication Tag",
    "encoding": "base64"
  },
  "data": "Base64 編碼後之加密 JSON 字串"
}
```

3.3 明文 payload 範例

```
{
  "source_record_id": "SRC-20260622-001",
  "source_updated_at": "2026-06-22T09:12:00+08:00",
  "doctor_id": "A123456789",
  "doctor_name": "王小明醫師",
  "order_code": "[\"TMAHT1\",\"TMAHT2\"]",
  "patient_national_id": "A123456789",
  "dob": "1980-01-01",
  "patient_name": "王小明",
  "mobile_phone": "0912345678",
  "remarks": "近兩週睡不好且久坐"
}
```

欄位註解（含必填）：

欄位	必填	說明
`source_record_id`	是	HIS 廠商系統內的推送唯一識別碼，每次推送請保持唯一。
`source_updated_at`	是	來源資料最後更新時間，建議使用 ISO 8601 格式。
`doctor_id`	是	HIS 廠商端使用的醫師識別碼，建議帶醫師身分證字號，用來對應內部帳號。
`doctor_name`	是	醫師姓名，供顯示與對帳使用。

欄位	必填	說明
`order_code`	是	醫令代碼，可為單一代碼或 JSON 陣列字串。
`patient_national_id`	是	病患身分證字號或統編。
`dob`	是	病患生日，格式 `YYYY-MM-DD`。
`patient_name`	是	病患姓名。
`mobile_phone`	否	聯絡手機。
`remarks`	否	備註或補充說明。

3.4 `curl` 範例

把前面產生的 envelope JSON 放進 `BODY` 變數，再送出。

```
curl -X POST "$HIS_TEST_BASE_URL/api/v1/his/medical-order/push" \
  -H "Authorization: Bearer $ACCESS_TOKEN" \
  -H "Content-Type: application/json" \
  -H "X-Timestamp: $X_TIMESTAMP" \
  -H "X-Nonce: $X_NONCE" \
  -H "X-Signature: $X_SIGNATURE" \
  -H "X-Request-Id: $X_REQUEST_ID" \
  --data-raw "$BODY"
```

3.5 Response（成功 200）

```
{
  "status": "ok",
  "received_at": "2026-06-22T09:12:01.123456Z",
  "source_record_id": "SRC-20260622-001"
}
```

3.6 Response（409：一年內已有處方紀錄）

如果同一位病患在一年內已經開立過相同類型的處方，系統會回傳 `409 Conflict`：

```
{
  "detail": "以下處方在一年內已有紀錄（同身分證 A123456789）：運動處方（最後一次 2026-06-22）、營養處方（最後一次 2026-06-22）、社會處方（最後一次 2026-06-22）、情緒調適處方（最後一次 2026-06-22）。"
}
```

> 這是目前系統的業務規則之一，表示該次醫令已命中一年內重複開立限制。

4. 健檢同步 API

4.1 端點

- Method：`POST`
- Path：`/api/v1/his/health/batch-sync`

4.2 Request Body

送出的 body 一樣必須是加密後的 envelope：

```
{
  "encryption": {
    "algorithm": "AES-256-GCM",
    "key_id": "$HIS_VENDOR_ACCOUNT",
    "iv": "Base64 編碼後之 IV / Nonce",
    "tag": "Base64 編碼後之 Authentication Tag",
    "encoding": "base64"
  },
  "data": "Base64 編碼後之加密 JSON 字串"
}
```

4.3 明文 payload 範例

```
{
  "records": [
    {
      "patient_national_id": "A123456789",
      "id_type": "本國人",
      "name": "王小明",
      "gender": "男性",
      "dob": "1980-01-01",
      "phone": "0912345678",
      "address": "台北市信義區信義路五段7號",
      "email": "vendor@example.com",
      "contact_name_relation": "王小華/配偶",
      "fetal_info": null,
      "parent_patient_national_id": null,
      "vaccine_code": null,
      "vaccine_dose": null,
      "vaccine_lot_number": null,
      "lot_type": null,
      "vaccine_brand": null,
      "package_type": null,
      "stock_quantity": null,
      "reserved_quantity": null,
      "consumed_quantity": null,
      "remaining_quantity": null,
      "identity_category": "一般民眾",
      "treatment_fee_remark": "門診追蹤",
```

```
        "exam_type": "成人健檢",
        "blood_draw_date": "2026-06-22",
        "igg_result": null,
        "igm_result": null,
        "is_prenatal_check": false,
        "hbsag": null,
        "hbeag_result": null,
        "hbv_viral_load": null,
        "expected_delivery_date": null,
        "anti_hcv": null,
        "order_code": "S12345678",
        "report_sequence": "RPT-20260622061131",
        "exam_item_name": "成人預防保健",
        "exam_method": "抽血與理學檢查",
        "exam_result_value": "追蹤中",
        "exam_positive_negative": null,
        "unit": null,
        "reference_range": null,
        "report_image": null,
        "pathology_diagnosis": null,
        "specimen_method_source": "靜脈採血",
        "treatment_site": "家庭醫學科",
        "result_remark": "近兩週睡不好且久坐",
        "radiation_dose": null,
        "height": 168.0,
        "weight": 72.5,
        "systolic_blood_pressure": 138,
        "diastolic_blood_pressure": 92,
        "abdominal_circumference": 84.5,
        "bmi": 25.7,
        "protein_urine": "negative",
        "glucose_ac": 108.0,
        "cholesterol": 205.0,
        "triglyceride": 168.0,
        "ldl_c": 128.0,
        "hdl_c": 48.0,
        "ast": 22,
        "alt": 28,
        "creatinine": 0.92,
        "egfr": 96.4,
        "uric_acid": 6.1,
        "renal_exam_period": "年度",
        "smoking": "否",
        "alcohol": "偶爾",
        "betel_nut": "否",
        "exercise": "不規律",
        "disease_history": "高血壓",
        "hepatitis_bc_history": "無",
        "depression_q1": "偶爾",
        "depression_q2": "偶爾",
        "depression_result": "需關注",
        "bp_result_suggestion": "血壓偏高，建議規律量測",
        "glucose_result_suggestion": "血糖偏高，建議控制飲食",
        "lipid_result_suggestion": "血脂偏高，建議增加運動",
```

```
        "renal_result_suggestion": "腎功能正常",
        "liver_result_suggestion": "肝功能正常",
        "metabolic_syndrome_result": "需持續追蹤",
        "hbv_result_suggestion": null,
        "hcv_result_suggestion": null,
        "risk_coronary_heart_disease": 0.08,
        "risk_diabetes": 0.31,
        "risk_hypertension": 0.26,
        "risk_stroke": 0.14,
        "risk_cardiovascular_event": 0.11,
        "update_time": "2026-06-22T06:11:31.936523Z"
      }
    ]
  }
```

欄位註解（含必填）：

- `records`：必填，本次批次同步的健檢資料陣列，至少 1 筆。
- 每筆 `record` 的欄位 key 都要保留，不可省略。
- 必填且不可為 `null`：`patient_national_id`、`name`、`dob`、`update_time`。
- 其餘欄位：key 仍然必填，但 value 可以是 `null`。

欄位	必填	可為 null	說明
`patient_national_id`	是	否	病患身分證字號或統編。
`name`	是	否	病患姓名。
`dob`	是	否	出生日期，格式 `YYYY-MM-DD`。
`update_time`	是	否	此筆資料最後更新時間。
其他欄位	是（key）	是	依實際資料填寫；若無值請傳 `null`，但 key 不能省略。

4.4 `curl` 範例

把前面產生的 envelope JSON 放進 `BODY` 變數，再送出。

```
curl -X POST "$HIS_TEST_BASE_URL/api/v1/his/health/batch-sync" \
  -H "Authorization: Bearer $ACCESS_TOKEN" \
  -H "Content-Type: application/json" \
  -H "X-Timestamp: $X_TIMESTAMP" \
  -H "X-Nonce: $X_NONCE" \
  -H "X-Signature: $X_SIGNATURE" \
  -H "X-Request-Id: $X_REQUEST_ID" \
  --data-raw "$BODY"
```

4.5 Response（成功 200）

```
{
  "status": "ok",
  "generated_at": "2026-06-22T09:12:01.123456Z",
  "total_count": 1,
  "records": [
    {
      "patient_national_id": "A123456789"
    }
  ]
}
```

5. 常見注意事項

- `Authorization` 需要使用登入成功後拿到的 `access_token`
- `X-Timestamp`、`X-Nonce`、`X-Signature`、`X-Request-Id` 都要帶
- `curl` 建議使用 `--data-binary @file.json`，不要直接把長 JSON 塞進命令列字串
- 健檢同步 response 目前只回傳最小必要摘要，不再回傳完整健檢內容
- `base64` 只用在 envelope 的 `data` 欄位編碼，不代表內容未加密