



OpenAPI Authorization & Authentication

Open API授權方法說明 新版

v1 & v2 method

2020.09.09 Jin Chou



基本說明	3
串接前評估	3
問題排除方法	3
其他相關連結	3
賣場授權方法	4
授權說明	4
賣場授權v1&v2方法說明	4
v1授權教學	4
v1授權網址組成說明	4
v1授權網址組成範例	5
v1授權網址組成步驟	5
v1授權逾時邏輯	5
Code demo	6
v1驗證請求說明	7
請求範例	7
Step1.建立簽名基本字串	7
Step2.通訊協定	7
Step3.HTTP呼叫內容	7
Step4.準備金鑰	7
Step5.簽名加密	7
v2授權教學	9
v2授權網址組成說明	9
v2授權網址組成範例	9
v2授權網址組成步驟	10
v2授權逾時邏輯	10
v2驗證請求說明	11
呼叫方法	11
Step1.建立簽名基本字串	11
Step2.簽名加密	12
Get Access_token	12
Refresh Access_token	13
取得Shop ID方法	15



基本說明

串接前評估

若欲查看Open API平台相關介紹和API功能，請查看此[說明事項](#)。
或請貴司的工程師參考此[平台文件](#)，以評估您預期欲使用的功能是否可透過本API串接達成。

問題排除方法

若您串接時遇到問題，建議可先[查找F&Q](#)。
若仍無法解決您的問題，請來信openapi@support.shopee.com或是聯繫您的客戶經理且提供以下資訊。

- 對應的問題描述：
- 所使用的對應環境：
- 所使用的對應API接口：
- 所使用的對應request body及所屬站點：
- 調用接口之後所得到的對應返回req_id（具體時間）及對應返回的信息（錯誤代碼和錯誤信息）：

其他相關連結

賣家幫助中心

- [API串接功能說明](#)
- [API賣場授權說明事項](#) (此文件下載位置)

Open API Platform

- [API所有文件](#) (English Only)

其他

- [F&Q](#)



賣場授權方法

授權說明

賣場授權v1&v2方法說明

為了保護Shopee OpenPlatform免受潛在的漏洞影響並更好地保護開發人員的機密性，OpenAPI 2.0（v2）引入了更新方法來授權商店和驗證API請求。

v2方法只能應用於v2 API，而v1方法只能應用於v1 API。
可透過呼叫網址判斷該API是v1或v2，如下表：

API呼叫網址
https://partner.shopeemobile.com/api/v1/item/add
https://partner.shopeemobile.com/api/v2/shop/cancel_auth_partner

v1授權教學

v1授權網址組成說明

開發人員按照以下規則生成授權連結，連結提供給賣場營運人員進行授權。授權完畢後即可透過API對店鋪進行操作。

開發人員應使用Partner ID、Key提供的測試憑據和測試車間信息來模擬此過程。
授權連結組成需包含一個固定的授權URL和必需的參數。**取消授權作法流程同賣場授權流程。**

授權網址	
https://partner.shopeemobile.com/api/v1/shop/auth_partner	正式環境
https://partner.uat.shopeemobile.com/api/v1/shop/auth_partner	測試環境

取消授權網址	
https://partner.shopeemobile.com/api/v1/shop/cancel_auth_partner	正式環境
https://partner.uat.shopeemobile.com/api/v1/shop/cancel_auth_partner	測試環境



授權使用參數

id	Partner ID可點選右上角console後選擇APP，則可在APP List中查看
token	請按照要求產生token，參考下方"token組成範例"
redirect	僅在授權成功後導向的頁面，可以使用任何網址(e.g.公司網站、賣場網址)

v1授權網址組成範例

```
https://partner.shopeemobile.com/api/v1/shop/auth_partner?id=xxxxxx&token=xxxxxxxxxxxx&redirect=xxxxxx.com
```

v1授權網址組成步驟

1.組成基本字串

Partner key和redirect url為構成組成的重點，組成範例Partner key+redirect url:

```
9b754aba01a5d719cc70c57782941ae6ff90fcc687282908ee480a364901d181https://www.google.com
```

2.加密字串

依上一步驟將字串結合後需要透過SHA256 hashing algorithm(not HMAC)加密，加密後則為Token。加密後範例如下:

```
815d97b3a582e2e39957545a0c8c3da63d2466b902a32956ddd7a784badccdb8
```

3.組成授權網址

接著按照上述"授權網址組成範例"的規則組成授權網址

```
https://partner.shopeemobile.com/api/v1/shop/auth_partner?id=70148&token=8e23ec43c9ef372d998797c029b48545a21b01e876a58b51b02ad1759c8bc4de&redirect=https%3A%2F%2Fwww.baidu.com
```

v1授權逾時邏輯

透過任何瀏覽器進入賣場授權網址(如上述"授權網址組成範例")後，您將會被重新導向，重新導向的網址為亂數產生，而亂數產生後的網址需要在3分鐘內完成賣場授權或取消賣場授權，若未在3分鐘內完成將會得到錯誤訊息"operation overtime"，需要重新進入賣場授權網址，無法直接重新整理使用亂數產生後的網址登入。



Code demo

python:

```
import hashlib
def cal_token(redirect_url, partner_key):
    base_string = partner_key + redirect_url
    token = hashlib.sha256(base_string).hexdigest() # note: not HMAC
    return token
```

golang:

```
package main
import (
    "crypto/sha256"
    "encoding/hex"
    "fmt"
)
func CalToken(redirectURL, partnerKey string) (result string) {
    baseStr := partnerKey + redirectURL
    h := sha256.New()
    h.Write([]byte(baseStr))
    result = hex.EncodeToString(h.Sum(nil))
    return result
}
```

java::

```
public static String calToken(String redirectURL, String partnerKey) {
    String baseStr = partnerKey + redirectURL;
    return org.apache.commons.codec.digest.DigestUtils.sha256Hex(baseStr);
}
```



v1 驗證請求說明

此為呼叫API時須使用的加密授權驗證。

以下教學同[Shopee Open Platform - Authenticating a Request](#)。

請求範例

這一部分將說明如何創建HMAC-SHA256加密簽名，針對HTTP請求。

請求方式為POST，以下範例使用API:<https://partner.shopeemobile.com/api/v1/orders/detail>

```
POST /api/v1/orders/detail HTTP/1.1
Host: partner.shopeemobile.com
Content-Type: application/json
Content-Length: 89
Authorization: b37c061daf2fcfa2baffe7539110938be5b7525041c147e78ad8afa78cc1a72d

{"ordersn": "160726152598865", "shopid": 61299, "partner_id": 1, "timestamp": 1470198856}
```

Step1. 建立簽名基本字串

基本字串產生規則如下：

API URL | Request

基本字串範例如下：

```
https://partner.shopeemobile.com/api/v1/orders/detail{"ordersn":"160726152598865","shopid":61299,"partner_id":1,"timestamp":1470198856}
```

Step2. 通訊協定

當呼叫API及產生加密簽名時，使用正確的通訊協定非常重要！

請確保使用的通訊協定為[https://](#)或[http://](#)。

Step3. HTTP呼叫內容

當呼叫時，您的HTTP呼叫格式該使用raw，HTTP header使用 [Content-Type: application/json](#)

Request範例如下：

```
{"ordersn": "160726152598865", "shopid": 61299, "partner_id": 1, "timestamp": 1470198856}
```

Step4. 準備金鑰

金鑰為呼叫API的重要關鍵，請勿隨意對外透漏您的金鑰。

Step5. 簽名加密

最後，通過將第1步產生的基本字串和金鑰傳遞給[HMAC-SHA256 hashing algorithm](#)加密簽名後即可使用在呼叫API時使用。



加密後Authorization範例如下:

b37c061daf2fcfa2baffe7539110938be5b7525041c147e78ad8afa78cc1a72d



v2授權教學

v2授權網址組成說明

開發人員按照以下規則生成授權連結，連結提供給賣場營運人員進行授權。授權完畢後即可透過API對店鋪進行操作。**取消授權作法流程同賣場授權流程。**

授權網址	
https://partner.shopeemobile.com/api/v2/shop/auth_partner	正式環境
https://partner.uat.shopeemobile.com/api/v2/shop/auth_partner	測試環境

取消授權網址	
https://partner.shopeemobile.com/api/v2/shop/cancel_auth_partner	正式環境
https://partner.uat.shopeemobile.com/api/v2/shop/cancel_auth_partner	測試環境

授權使用參數			
欄位	類型	必要	說明
partner_id	int	yes	Partner ID可點選右上角console後選擇APP，則可在APP List中查看
redirect	string	yes	僅在授權成功後導向的頁面，可以使用任何網址 (e.g.公司網站、賣場網址)
sign	string	yes	依Shopee規則產生的字串
timestamp	int	yes	需要與sign字串使用的時間戳保持一致。 5分鐘後過期。

返回內容			
欄位	類型	必要	說明
code	string	yes	用於獲取access_token和refresh_token。一次性使用有效期為10分鐘。
shop_id	int	yes	完成授權的shop_id。

v2授權網址組成範例

本部分說明如何生成用於賣場授權的HMAC-SHA256。



v2授權網址組成步驟

1.組成sign的規則

按照此順序，以partner_id , api path , timestamp組成sign。

```
10090/api/v2/shop/auth_partner1594897040
```

2.加密字串

依上一步驟將字串結合後需要透過SHA256 hashing algorithm(not HMAC)加密，加密後則為Token。加密後範例如下：

```
90c12d3932f3826f0c72242e1ec6492eec9a1298658f41f7a9469664801c4e5a
```

3.組成授權網址

接著按照上述"授權網址組成範例"的規則組成授權網址

```
https://partner.uat.shopeemobile.com/api/v2/shop/auth_partner?partner_id=10090&redirect=https://www.baidu.com/&timestamp=1594897040&sign=90c12d3932f3826f0c72242e1ec6492eec9a1298658f41f7a9469664801c4e5a
```

4.完成授權

依照上述網址完成賣場授權後，將重新導至此網址：(xxx表示隨機或以該賣場shop id為主)

```
https://www.baidu.com/?code=xxxxxxxx&shop_id=xxxxxx
```

v2授權逾時邏輯

授權網址會在5分鐘內過期(因為Timestamp過期)，所以開發人員必須重新生成一個新的授權網址。

若進入授權網址後，未在3分鐘內登入賣場帳號、密碼，將會得到錯誤訊息"operation overtime"，需要重新進入賣場授權網址，而不是重新整理該頁面。



v2驗證請求說明

除非指定對某些特定的API使用HTTP / FORM（在API文檔中有說明），否則對所有其餘的API使用HTTP / JSON。

以下教學同[OpenAPI 2.0 Authorization & Authentication](#)。

呼叫方法

支援POST、GET

使用參數			
欄位	類型	必要	說明
partner_id	int	yes	Partner ID可點選右上角console後選擇APP，則可在APP List中查看
timestamp	int	yes	所有請求都必須包含。5分鐘後過期
access_token	string	no	將access_token用作某些API的通用請求參數，在過期前可以多次使用，有效期為4小時 要調用Get Access_token API和刷新Access_token API，可以跳過此參數
sign	string	yes	依Shopee規則產生的字串
shop_id	int	no	賣場Shop id

返回內容			
欄位	類型	必要	說明
request_id	string	yes	用於提供給Shopee查詢呼叫紀錄
error	string	yes	錯誤訊息，若無錯誤則返回空值
message	string	no	詳細錯誤訊息說明
response	object	no	返回內容

Step1.建立簽名基本字串

本部分說明如何為API請求用的驗證生成HMAC-SHA256。

按照此順序，使用partner_id，api path，timestamp，access_token 和shop_id 組成sign(要調用Get Access_token API和Refresh Access_token API，可以在符號基礎字符串中跳過access_token)。



Step2.簽名加密

將Sign和Partner key以HMAC-SHA256 hashing algorithm來加密。

Python Code Demo:

```
#!/usr/bin/env python
# encoding: utf-8

import hmac
import time
import requests
import hashlib

timest = int(time.time())
host = "https://partner.shopeemobile.com"
path = "/api/v2/item/get"
access_token = "random string"
shop_id = 209920
partner_id = 80001
partner_key = "test...."
base_string = "%s%s%s%s%s"%(partner_id, path, timest, access_token, shop_id)
sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()

####call api
url = host + path + "?partner_id=%s&shop_id=%s&timestamp=%s&access_token=%s&sign=%s"%(partner_id, shop_id, timest, access_token, sign)
body = {"partner_id":partner_id, "shop_id": shop_id, "item_id": 1}
headers = { "Content-Type": "application/json"}
resp = requests.post(url, json=body, headers=headers)
```

Get Access_token

API: <https://partner.shopeemobile.com/api/v2/auth/token/get>

Method: Post

Description: 使用此API和代碼來獲取access_token和refresh_token。

使用參數

欄位	類型	必要	說明
code	string	yes	賣場授權後, 重新導向URL中的代碼 。為一次性使用, 有效期為10分鐘
partner_id	int	yes	開發者APP的Partner id
shop_id	int	yes	賣場Shop id

返回內容

欄位	類型	必要	說明
----	----	----	----



request_id	string	yes	用於提供給Shopee查詢呼叫紀錄
error	string	yes	錯誤訊息, 若無錯誤則返回空值
refresh_token	string	yes	使用refresh_token獲取新的access_token。一次性使用, 有效期為30天
access_token	string	yes	將access_token用作某些API的通用請求參數。有效多次使用, 有效期為4小時
expire_in	int	yes	access_token過期時間, 單位為秒

Python Code Demo:

```
import time
import json
import hmac
import hashlib

def get_token(code, partner_id, partner_key, shop_id):
    timest = int(time.time())
    body = {"code": code, "shop_id": shop_id, "partner_id": partner_id}
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/token/get"
    base_string = "%s%s%s%s"%(partner_id, path, timest, shop_id)
    sign = hmac.new( partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&shop_id=%s&timestamp=%s&sign=%s"%(partner_id, shop_id, timest, sign)

    headers = { "Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)
    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token
```

Refresh Access_token

API: https://partner.shopeemobile.com/api/v2/auth/access_token/get

Method: Post

Description: 使用此API和代碼來重新獲取access_token當access_token過期後。

使用參數

欄位	類型	必要	說明
refresh_token	string	yes	使用refresh_token獲取新的access_token。一次性使用, 有效期為30天。



partner_id	int	yes	開發者APP的Partner id
shop_id	int	yes	賣場Shop id

返回內容

欄位	類型	必要	說明
request_id	string	yes	用於提供給Shopee查詢呼叫紀錄
error	string	yes	錯誤訊息，若無錯誤則返回空值
refresh_token	string	yes	使用refresh_token獲取新的access_token。一次性使用，有效期為30天
access_token	string	yes	將access_token用作某些API的通用請求參數。有效多次使用，有效期為4小時
expire_in	int	yes	access_token過期時間，單位為秒
partner_id	int	yes	開發者APP的Partner id
shop_id	int	yes	賣場Shop id

Python Code Demo:

```
#!/usr/bin/env python
# encoding: utf-8

import time
import json
import hmac
import hashlib
import requests

def get_access_token(shop_id, partner_id, partner_key, refresh_token):
    timest = int(time.time())
    host = "https://partner.shopeemobile.com"
    path = "/api/v2/auth/access_token/get"
    body = {"shop_id": shop_id, "refresh_token": refresh_token}
    base_string = "%s%s%s"%(partner_id, path, timest)
    sign = hmac.new(partner_key, base_string, hashlib.sha256).hexdigest()
    url = host + path + "?partner_id=%s&timest=%s&sign=%s"%(partner_id, timest, sign)

    headers = {"Content-Type": "application/json"}
    resp = requests.post(url, json=body, headers=headers)

    ret = json.loads(resp.content)
    access_token = ret.get("access_token")
    new_refresh_token = ret.get("refresh_token")
    return access_token, new_refresh_token
```



取得Shop ID方法

授權後欲查看Shop ID可參考以下方法:

- 一經賣家授權完成後，開發者的信箱將會收到授權通知信，信件內部將提及Shop ID。**(測試環境不支援)**
- Shop ID將會出現在授權完成後跳轉的頁面網址中，例如在組成賣家授權網址時，填入Redirect Uri 為https://www.google.com/，則一經授權網址則會加上Shop ID，如下圖所示:



- 如果啟用了Webhook功能，JSON消息將被推送到開發者的Callback URL。**(測試環境不支援)**
- 可透過`shopee.public.GetShopsByPartner`獲取該Partner id下的所有Shop id。

