


OPEN API TEST GUIDE


INTRODUCTION

This document is intended to guide merchants or third parties through the various services of the Payment Gateway. The Payment Gateway's interfaces respond successfully when there is a valid Bearer authorization token specified in the request as well as a valid signature(sign). Thus, tokens can be registered by invoking the [token](#) interface, by specifying the [X-APP-Key](#) and [appSecret](#) in the request header and request body respectively.

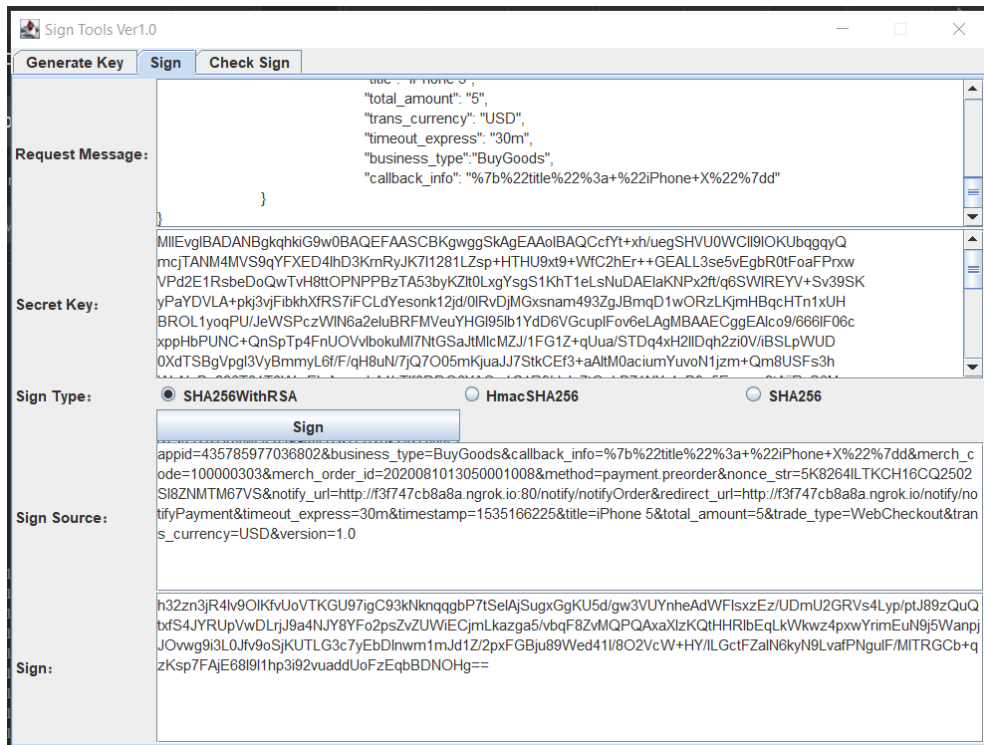
Also, request bodies may be signed with either SHA256WithRSA or HmacSHA256 algorithm depending on the interface being invoked. You can verify the signature with the [sign tool](#) to make sure everything goes right. Please follow the steps below to generate a signature using the sign tool:

- Extract the SignTool.zip file
- Run Tools.jar (please ensure you have java already installed on the host PC)

 [run.bat](#)

 [Tools.jar](#)

- Navigate to the Sign Tab
- Copy and paste the request body into the Request Message text field
- Copy and paste the appropriate [sign key](#) for the interface being invoked into the Secret key text field. (please provide the private key if the algorithm is SHA256WithRSA)
- Select the Sign Type and click the Sign button
- Copy the signature from the Sign text field



USE CASES

- DISBURSEMENT

This function is suitable for enterprises or merchants hoping to disburse salary or transfer funds to consumers. When a merchant calls the [Disbursement](#) interface of the Payment Gateway, the Gateway verifies that the interface is authorized for the merchant and the merchant has sufficient balance, the Payment Gateway will immediately complete the funds merchant-to-customer transfer operation and return the transfer result in the interface response. The payee customer must be registered in the wallet, and the status is normal; Before calling this interface, the [KYC lookup](#) API needs to be called for verification. Also, the merchant needs to be subscribed to the (Disbursement) product.

Please Note: The payment interface is real-time fund transfer. The caller must ensure that the payee information is correct, otherwise the funds of payer will be lost.

DISBURSEMENT PROCESS FLOW

1. The 3rd party system calls the [KYC lookup](#) interface for payee MSISDN verification.
2. The customer then confirms the KYC and proceed.
3. The 3rd party system then invokes the [Disbursement](#) interface of the Payment Gateway for transferring funds to a consumer from an enterprise.
4. After having disbursed funds between an organization and a consumer, the Payment Gateway responses the result of disbursement.
5. If the 3rd party system has not received the result of invoking disbursement until the connection is timeout, the 3rd Party system can also invoke the [Query order](#) interface to get the result of disbursement by Merchant Order ID.
6. The Payment Gateway returns the disbursement result of the order.

- **WEBCHECKOUT CREATE ORDER**

The webcheckout [Create order](#) is the interface suitable for merchants or third parties for the collection of funds or payments from consumers. The Checkout page can be accessed from the merchant's web page or mobile application. Merchants can receive payments by creating an order and then redirecting the consumer to the checkout page to confirm payment.

Please Note: The appropriate trade_type for creating a webcheckout order is "WebCheckout".

WEBCHECKOUT PROCESS FLOW

1. The user completes an order and clicks the checkout button on the merchant's web page or mobile application.
2. The merchant server receives the order request and calls the Payment Gateway's unified order interface([Create order](#)) to create a prepaid order.
3. The merchant server signs the webcheckout parameters; appid, merch_code, nonce_str (from create order response), prepay_id (from create order response) and timestamp.
4. The merchant constructs a [webcheckout URL](#) with the appropriate values for appid, merch_code, nonce_str, prepay_id, timestamp and sign.
5. The merchant then redirects the user to the [webcheckout URL](#) for the user to confirm and complete the payment.

6. The user enters his/her MSISDN and verification code, and then enters the PIN to confirm the payment.
7. If the payment is successful, the Payment Gateway asynchronously notifies the merchant server of the payment result(only if the notify_url was specified in the [Create order](#) request). The user may be redirected back to the merchant's web page by clicking "Return to the merchant" button(only if the redirect_url was specified in the [Create order](#) request)
8. The merchant server determines whether the payment result notification from Payment Gateway is received. If not, the merchant server calls the Payment Gateway order inquiry interface ([Query order](#)) to confirm the order status.
9. The merchant displays the final order payment result to the user.
10. If a merchant wishes to cancel a pending order, it can be done by invoking the Payment Gateway's [Close order](#) interface.
11. The merchant server can also request for a refund to consumers by invoking the [Refund](#) interface.
12. The merchant server may also invoke the [Query refund](#) interface to inquire the status of the refund request by the merchant order ID and refund request number.

APPENDIX

ACCESS URLS

OpenAPI URL

<https://124.71.15.229:32340/apiaccess/sandboxgroup/payment/gateway>

Example

<https://124.71.15.229:32340/apiaccess/sandboxgroup/payment/gateway/payment/v1/merchant/preOrder>

WebCheckOut URL

<https://124.71.15.229:32345/payment/web/paygate>

Example

https://124.71.15.229:32345/payment/web/paygate?appid=435785977036802&merch_code=100000303&nonce_str=c23416cfd7234ae0bdcd2d2bb8f2dc3b&prepay_id=00110120081012472654720725cc81473677648068001×tamp=1535166225&sign=99da1f758225de2830bcd6f155fe8950dcb98c74ee00355f7f7258ff043f79b5&sign_type=HmacSHA256&version=1.0&trade_type=WebCheckout

Token URL

<https://124.71.15.229:32340/apiaccess/sandboxgroup/payment/gateway/payment/v1/token>

API DOCUMENT URL

<http://159.138.166.39/api.html#tag/Payment-Management>

TEST ACCOUNT

ACCESS APP

X-APP-Key : bbbb5d11-f106-40c0-a578-64c991f46fe3

appSecret : 3c58b56515596b3d1d7245e4c6ff4576

TEST IDENTITY

merch_code: 100000303

appId: 435785977036802

Customer MSISDN: 8615000000001

MSISDN Verification Code: 123456

PIN: 2468

SIGN KEYS

SHA256WithRSA

PUBLIC KEY:

MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAnH2LfsYf7noEh1VNF
giJfSDiIG6oKskJnI0wDTODFUvamBVxA+JYQ9yq50ciSuyNdvNS2bKfh0x1Pcbffln
wtoRK/vhhACy97HubxIG0dLRaGhT68cFT3dhNUbG3g6EME7x/LbTjzTzwc0wO
d28imZbdC8YGLIEtSoU9Xi7DbgwBJWijt8dn7f6uklpURGFfkr9/Uisj2mA1SwPq
ZI974xYm5IV30Uu4hQi3WHrKJ5Ndo3f9JUbw4zBsbJ2puPd2YCOZqg9cDkcyo
5hwanB059cVBwUTi9cqKj1PyXlkj3M1pTemtnpbgURTFXrmBxpfeSG9WHQ+IR
nLqSBaL+niwIDAQAB

PRIVATE KEY:

MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAKgwggSkAgEAAoIBAQCcfYt+xh/uegS
HVU0WCil9I0KUbqgqyQmcjTANM4MVS9qYFXED4lhD3KrnRyJK711281LZsp+H
THU9xt9+WfC2hEr++GEALL3se5vEgbR0tFoaFPrxwVPd2E1RsbeDoQwTvH8ttO
PNPPBzTA53byKZlt0LxgYsgS1KhT1eLsNuDAElaKNPx2ft/q6SWIREYV+Sv39SKyP
aYDVLA+pkj3vjFibkhXfRS7iFCLdYesonk12jd/OIRvDjMGxsnam493ZgJBmqD1wO
RzLKjmHBqcHTn1xUHBROL1yoqPU/JeWSPczWIN6a2eluBRFMVeuYHGI95Ib1Y
dD6VGcupIFov6eLAgMBAAECggEAlco9/666IF06cxppHbPUNC+QnSpTp4FnUO
VvlbokuMI7NtGSaJtMlcMZJ/1FG1Z+qUua/STDq4xH2IIDqh2zi0V/iBSLpWUD0X
dTSBgVpJ3VyBmmyL6f/F/qH8uN/7jQ7005mKjuaJJ7StkCEf3+aAltM0aciumYu
voN1jzm+Qm8USFs3hWvNcBn286T01TOWwEloAr+nnhA1bTif0DDO0YAO+1G
1R9LkJu7tOqLB71NYr4eB9u5E+yyrq8tAiiRxS6MuYBLIU5peRXQVTJzV4pL4uD
dFORaIA/eU3NPWBEhAOr0Jvw8TeN+HOX27iKYcgdpMKsisWcYT+ASmoeQKBg

QDZrdk6dj73y63dU10laLJG1pET2J84tptFEHweLwyAoIMtVnZbATLRslhSstr9F
m8fLFpLCWsHDRhNSZL6Ult6WUa2bFTEqthaRrXNFap816bCghbhzYkQa6AM0
dNB1bcO9cTzrA7UQVqc9Js0EO6LahhxEJsHgkHBeO42DjpylwKBgQC4ChqLfk0
mwV9xvOJ0Wngr4SS0eqLnKomqLYtaXkkEvZsLZEZliVDRRE7ytSvbwqmHj3HRG
Kik+FQ8HDroJ7hmMAVTnhykHWnKg6gZSlprTv7hxlL8aWWMyWmuaY5kFy77
1AUSi6L1E/qT85RS9UYdv7BVr75smzDLEvFw7mn1LQKBgQCrpZirrt5MS4HYCo
pLvTMHBN71hVVYRTX+qdcVfggfzXJi59EARduq2cdFXt+rvmr1a+XvUYfh0YOePi
v108dqnMdTxey5//YGiceF5IW/h9Vpf2dHuukf154nPbWWLk3hMA+UDdLLsJU
hOwi/b7KKoKiRT4TP0PjE9PTyDdikFQKBgQCc6ZhqWLxLD7hV3Crj84LEPsixzGH
6YI1ReFV5Lixpi1OXKd7PkX1Zqjr5urhzPYyXIRkAo1qeVAUICkxaTRShBz2hzd6Q
c3CKfB35h6GkarFFQp+BUrc95lQaJVMB2w6sDHG9yFwfaFG5gYPmSo4EQjCdf
4Q0VyPuE388HmM3QKBgEvZVYdnz9RN3+xPkS+Qtn6K9aK+bcJGzrmu9F1uKP
91bMnoQasAM/N0iMm+krUGiBFNfwIzbGzGT7oqxKNDgcnLQrtLlOoAeCQVnc
hCBHh7rvVHeP9OfQRzDp+EA+YQ6ibiHxy0nv+eQSNrb7Vlwfkn6vR1oZ34ckq1f
hC3hRdX

HmacSHA256

4JNxE4byUbslq4mAJymiSB3FUJ5I2Pbi0UGmYS3F1Q=